2011-09

Design and task analysis for a game-based shiphandling simulator using an open source game engine (DELTA3D)

de Moraes, Claudio Coreixas

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/5608

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DESIGN AND TASK ANALYSIS FOR A GAME-BASED SHIPHANDLING SIMULATOR USING AN OPEN SOURCE GAME ENGINE (DELTA3D)
by
Claudio Coreixas de Moraes
September 2011
Thesis Co-Advisors: Christian Darken, Anthony Ciavarelli
Second Reader: Roberto de Beauclair

Approved for public release; distribution is unlimited
This thesis was done at the MOVES Institute

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2011	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Design and Task Analysis for a Game-Based Shiphandling Simulator Using an Open Source Game Engine (DELTA3D)			**5. FUNDING NUMBERS**
6. AUTHOR(S) Claudio Coreixas de Moraes			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number: NPS.2011.0087-IR-EP7-A.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

This thesis addresses the need for a navigation and shiphandling game-based training system at naval academies. The Yard Patrol Simulator (YPSim) is an application designed to reduce the knowledge gap between classroom instruction and hands-on training onboard naval academy training boats (YPs). The goal was to develop a proof-of-concept game-based simulator that uses 3D graphics to replicate basic tasks executed onboard the YPs. Two missions were selected for a brief task analysis study to determine the design of the respective game scenario and requirements. The design process involved in building user interface, physics model, 3D models, and artificial intelligence actors are described in this work. For thesis purposes, YPSim was designed using the Brazilian Naval Academy's YP as a training framework development environment. Using a sample of the final end user population, we conducted a user acceptance study of proof-of-concept version of YPSim (v0.14) at the Brazilian Naval Academy. The findings in this work can be generalized to any other naval academy or institution where basic navigation and shiphandling instruction is provided. Initial results from a prototype implementation of YPSim at the Brazilian Naval Academy provided insights into the potential use of this training system.

14. SUBJECT TERMS			**15. NUMBER OF PAGES**
Simulator Design, Serious Games, Shiphandling Simulator, Training, Virtual Environments, Simulation, Open Source, Brazilian Navy			152
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

THIS PAGE INTENTIONALLY LEFT BLANK

DESIGN AND TASK ANALYSIS FOR A GAME-BASED SHIPHANDLING SIMULATOR USING AN OPEN SOURCE GAME ENGINE (DELTA3D)

Claudio Coreixas de Moraes
Lieutenant Commander, Brazilian Navy
B.S., Brazilian Naval Academy, 1998

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MODELING, VIRTUAL ENVIRONMENT, AND SIMULATION (MOVES)

from the

NAVAL POSTGRADUATE SCHOOL
September 2011

Author: Claudio Coreixas de Moraes

Approved by: Christian Darken
Thesis Co-Advisor

Anthony Ciavarelli
Thesis Co-Advisor

Roberto de Beauclair
Second Reader

Mathias Kolsch
Chair, Modeling, Virtual Environments, and Simulation Academic Committee

Peter J. Denning
Chair, Computer Science Academic Committee

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis addresses the need for a navigation and shiphandling game-based training system at naval academies. The Yard Patrol Simulator (YPSim) is an application designed to reduce the knowledge gap between classroom instruction and hands-on training onboard naval academy training boats (YPs). The goal was to develop a proof-of-concept game-based simulator that uses 3D graphics to replicate basic tasks executed onboard the YPs. Two missions were selected for a brief task analysis study to determine the design of the respective game scenario and requirements. The design process involved in building user interface, physics model, 3D models, and artificial intelligence actors are described in this work. For thesis purposes, YPSim was designed using the Brazilian Naval Academy's YP as a training framework development environment. Using a sample of the final end user population, we conducted a user acceptance study of proof-of-concept version of YPSim (v0.14) at the Brazilian Naval Academy. The findings in this work can be generalized to any other naval academy or institution where basic navigation and shiphandling instruction is provided. Initial results from a prototype implementation of YPSim at the Brazilian Naval Academy provided insights into the potential use of this training system.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

3D	Three Dimensional
AI	Artificial Intelligence
BNA	Brazilian Naval Academy
BRG	Bearing
COG	Course Over Ground
CTA	Cognitive Task Analysis
DAU	Defense Acquisition University
DOF	Degrees Of Freedom
EBL	Electronic Bearing Line
ETA	Estimated Time of Arrival
FMBS	Full Mission Bridge Simulator
GM	Game Manager
GOMS	Goals Operators Methods Selection
GPS	Global Positioning System
GUI	Graphical User Interface
HUD	Heads Up Display
HTA	Hierarchical Task Analysis
JOOD	Junior Officer of the Deck
LOD	Level Of Detail
LSI	*Laboratório de Sistemas Integrados* (Integrated Systems Laboratory)
NROTC	Naval Reserve Officer Training Corps
NPS	Naval Postgraduate School
OOD	Officer of the Deck
PC	Personal Computer
RPM	Revolutions Per Minute
SOG	Speed Over Ground
UDP	User Datagram Protocol
UNREP	Replenishment at Sea
SME	Subject Matter Expert

SWO	Surface Warfare Officer
SWOS	Surface Warfare Officers School
VRM	Variable Range Marker
YP	Yard Patrol

ACKNOWLEDGMENTS

I would like to thank every person who crossed my way during this journey. More specifically, to the one who crossed it every day: My beloved wife Carla.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. MOTIVATION

Most of my career as a Surface Warfare Officer (SWO) at the Brazilian Navy was dedicated to shiphandling and navigation instruction. Between 2004 and 2009, I was stationed as a deck officer and shiphandling instructor at the Brazilian Navy Tall Ship “Cisne Branco,” receiving around 300 midshipmen per year for instructions on board. In 2009, I was assigned Commanding Officer of one of the three Brazilian Naval Academy’s (BNA) Yard Patrol (YP) crafts, the “Guarda-Marinha Jansen” (U-11). For a period of one year, I was responsible for one-third of the hands-on training activity at the BNA, which was a fascinating experience. Among the many lessons learned during my job as an instructor, two are remarkable truths: nothing substitutes for a motivated student in the audience, and time will always be a constraint if you expect your student to really master a learning objective.

As a result of many observations during the several training missions accomplished on board, and conversations with my colleagues at the BNA’s YP fleet, we have noticed that, in general, midshipmen have a poor understanding of the practical tasks executed on board the YPs. The YP is a ship designed to play an important role as an afloat laboratory for the courses related to navigation, shiphandling and naval operations. It is a place for the experimentation and practice of concepts too abstract to learn without reinforcements coming from trial-and-error practice. However, my colleagues and I observed that a huge knowledge gap was present between lectures and procedures manuals, and the afloat lab, the YP. Without any intermediate steps providing the basic skills required to play the roles on board, midshipmen became easily overwhelmed by the huge amount of new information needed for every practice onboard. This overwhelming sensation affects the midshipman’s motivation and consumes the instructor’s time in teaching the very basics steps. Instead of focusing 100% of

the allocated time on hands-on training in the real learning objectives, the YP's instructors had to share time with minor details that consumed most of the restricted schedule.

A U.S. Navy lieutenant stationed as navigation instructor at BNA reported something similar when he was a midshipman at the United States Naval Academy (USNA). After leaving the BNA and becoming a student at NPS, I have also talked to U.S. Navy officers who reported the same issues when midshipmen at the USNA. The similarities presented in both learning frameworks of BNA and USNA pushed me towards the present work. This thesis represents an attempt to reach a proof of concept for a new tool intended to reduce the knowledge gap observed between classroom and hands–on training at the YPs.

Figure 1. Real and Virtual BNA's YP.

B. OBJECTIVE

The primary objective of this thesis is to provide the design and a proof-of-concept prototype of a PC game-based simulator for navigation and shiphandling tasks carried by a generic naval academy's midshipmen onboard YPs. The secondary objective is the ability to extend this application as an instructional resource in the classroom as a medium-scale simulator.

C. APPROACH

To achieve the proposed objectives, this thesis is conducted in four major steps: review, task analysis, development, and test with refinements.

First, using the BNA environment as an example, a broad introduction of navigation and shiphandling instruction is presented, along with a brief review of cognitive aspects of learning game-based systems and current simulations. The second step is towards the identification of onboard midshipmen activities that are interesting enough to be simulated in the game, called the YP Simulator (or shortened to "YPSim"). A task analysis of two selected missions is performed, providing instructional guidance to a requirements section of the thesis. Following these requirements, the third step is the development of the game platform, using Delta3D open source game engine. With a prototype version available, the fourth and final step is taken, leading to user acceptance test results at BNA. Based on feedback collected from the user study, software refinements are applied to the original code, ending the design cycle proposed for the scope of this thesis.

D. CHAPTER OUTLINE

1. Introduction

This chapter presents a brief explanation of the research problem and respective steps taken to address it.

2. Background

In the Background chapter, we first describe in details the current navigation and shiphandling instruction framework commonly adopted at naval academies using YPs. Secondly, a description of the cognitive process developed by midshipmen onboard the YP is given, providing a better understanding of the operational side of the problem. The next step was to provide a general idea of few current game-based simulations systems developed for similar training purposes, ending with a brief conclusion.

3. Missions Description and Cognitive Task Analysis

This chapter introduces the description of the basic training missions performed onboard the BNA YPs, the cognitive task analysis of two selected missions, including critical cues inventory and performance metrics.

4. Requirements Analysis

The YPSim's requirements analysis are presented in this chapter, defining costumers needs, objectives as a system and planned environment.

5. System Development

This chapter describes the research effort towards the development of YPSim functionality, models, and basic components that lead to YPSim v0.14, the proof of concept prototype version of the product.

6. YPSim Testing

The releases of testing versions of YPSim are documented, presenting the most important milestones for exposing the product to the public and collecting feedback for software refinements.

7. User Acceptance Survey

This chapter describes a user acceptance survey conducted at the BNA with 40 midshipmen who used YPSim in a lab room. We describe the method and results involved in this important study to evaluate system's design using end users feedback.

II. BACKGROUND

A. NAVIGATION AND SHIPHANDLING INSTRUCTIONS

Considered a mix of science and art (Barber, 2005), navigation and shiphandling are traditional aspects of any naval academy’s body of knowledge. Not only do naval academies offer courses related to these two subjects, but any other institution that deals with seamanship courses would include navigation and shiphandling in the curriculum. Various approaches are used to teach such topics and allow the students to reach the learning objectives. For the scope of this thesis, we are interested in a broad type of instruction provided by several naval academies worldwide, such as the United States Naval Academy (USNA) and the Brazilian Naval Academy (BNA). Both academies adopt a system that relies on classroom theory followed by hands-on training onboard. This thesis adopts the instruction model used at BNA, assuming that it is similar enough to USNA’s approach. We expect that all findings derived from this design can be generalized to any institution that uses a similar framework.

The first step towards navigation and shiphandling instruction happens inside the classroom. Instructors present the contents of the disciplines as they would any other regular core course, e.g., physics, calculus or programming, as a lecture. Inside a classroom is where most midshipmen will have their first exposure to subjects related to shiphandling. However, without having any experience aboard a ship, this theory becomes confusing and is often too abstract to be comprehended. As any other discipline that involves complex concepts, a solid understanding requires a lab.

The second step is the hands-on training. Through test and experimentation onboard the academy’s YPs, midshipmen solidify the abstract concepts presented in the classroom. The YPs are the academy’s navigation and shiphandling laboratories. Onboard the YPs—and playing roles as helmsman, navigator, lee-helmsman, radar operator, or the Officer of the Deck (OOD)—the

midshipmen are able to start developing their mental model, based on practical observation of real events related to the classroom theory. The mental model definition applied here is the one given by Hackos (1998) as something vague, individual and dynamic, which represents a collection of associations in people's minds, used to make connections between known and learned information.

Figure 2. Midshipmen during hands-on training at USNA (left) and BNA (right) YP bridges.

Although this classroom/YP combination seems powerful from an instructional perspective, some factors can limit its potential benefits, as listed below:

- Total time per midshipman aboard the YP is low.
- The fraction of the already small amount of YP time in certain key roles is low as well.
- The most important consideration may be the knowledge gap between the classroom and the hands-on training. A considerable amount of time and dedication is needed to familiarize a midshipman with the new controls, procedures, displays and organization onboard. Very often, onboard instructors spend a relatively large amount of time teaching very basic skills preparing a team of midshipmen to train for the objective tasks.

Over many years of use in institutions such as the USNA and the BNA, the learning framework using classroom and hands-on training has proven to be

very powerful. Ensigns who graduate from the USNA (receiving extensive training through the YPs) and reported to SWOs, demonstrate higher comprehensive shiphandling skills when compared to the NROTC Ensigns (Grassi, 2000). This thesis is not intended to provide any new model for the navigation and shiphandling instruction at naval academies, but to present a solution to mitigate or minimize the effects of one of the issues with the current framework. Providing a new instructional tool that is able to reduce the knowledge gap between the theory and practice would represent an important step towards improvements in the framework, reducing costs and loss of time during training. This thesis is about the design of a simulator for the very basic tasks executed during the hands-on training. Using a game-based system approach and open source libraries, a low-cost, easily accessible PC-simulator can be used to entertain, motivate and, at the same time, introduce the basics of the tasks conducted onboard the YP.

B. COGNITION ONBOARD THE YP

To better understand the issues here addressed, this section provides a brief analysis of the YP as a cognitive adaptive system, as proposed by Hutchins (1995). In his book “Cognition in the Wild” Dr. Hutchins makes a deep scientific exploration of the navigation tasks executed onboard a U.S. Navy ship, concluding the following:

> The conduct of the activity creates elements of representational structure that survive beyond the end of the task. These elements—the logbooks, pencil marks on charts, the quartermasters’ memories of the events—are the operational residua of the process. In this adaptive system, the media may be changed by the very processes that constitute the conduct of the activity. The operations of the navigation team produce a structured experience for the participants that contains opportunities for individual learning. As a consequence of their participation in the task performance, the quartermasters may acquire internal

organization that permits them to coordinate with the structure of their surroundings. In this way, learning can be seen as the propagation of organization through an adaptive system. (Hutchins, 1995)

The learning observed by Hutchins during the navigation is a constant process during hands-on training onboard the YP. The YP is designed for that exact purpose, and midshipmen—members of the team—are novices who interact with the system structure while the experimentation and observation transform their mental model of the task.

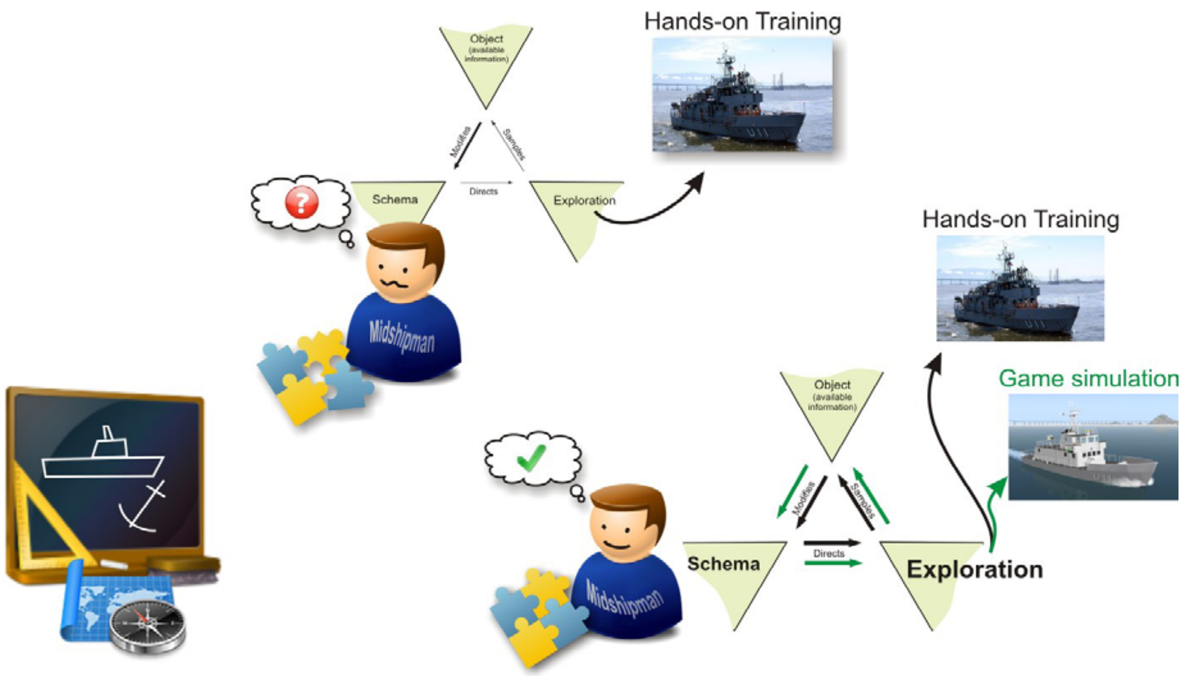


Figure 3. Neisser's Perceptual Cycle from a midshipmen perspective with YPSim incrementing the exploration step.

A typical example of the importance of experimentation in building a midshipman’s mental model is breaking the YP’s inertia. The instructors in the classroom present numbers and tables relating to a ship’s acceleration in general. These numbers will have a meaning to the midshipmen associating a given engine’s RPM and speeds. If the numbers are presented in a table, the understanding is very poor, since the numbers are rarely connected to any previously registered event. Although, when the instructor presents this

information using a graph, the student can easily correlate the acceleration with the slope of the curve—something that he has previous understanding with from math and physics courses. Going further, the instructor can present an animation of an accelerating ship model with those values, and the understanding will be even more solid and easier to correlate in a future experience. Now, imagine the student feeling the same acceleration onboard the YP. He/she has the opportunity to observe the OOD giving the RPM order to the lee-helmsman, the lee-helmsman actuating the controls, the RPM indicators increasing, the ship and the surrounding environment beginning to move, engine noise increasing, wake at the stern starts increasing, and so on. All this sensory information, known as cues, will be part of the mental model created in the midshipman's mind after the experience. As supported by Hutchins, the use of experimentation during the process facilitates the learning and building of a mental model of the task

The cognition involved in navigation and shiphandling tasks is not easily learned. It takes a considerable amount of time until midshipmen are able to master a given seamanship skill. As future officers who will be soon in leadership and decision-making positions, midshipmen have to gather a solid understanding of all sets of basic seamanship skills. After his observational study, Hutchins understood the complexity involved in learning navigation skills:

> The development of the practitioners themselves takes years. Through a career, a quartermaster gradually acquires the skills that are exercised in the performance of the job. Changes to the organization of the internal media that the quartermaster brings to the job take place more slowly than the changes to the states that the media support. That is, it takes longer to learn how to plot a fix, for example, than it does to plot a fix. But since most learning in this setting happens in the doing, the changes to internal media that permit them to be coordinated with external media happen in the same processes that bring the media into coordination with one another. The changes to the quartermasters' skills and the knowledge produced by this process are the mental residua of the process. (Hutchins, 1995)

Most of the midshipmen joining naval academies are going onboard a ship for the first time on the YPs. All the basic understandings of how a ship works—its organization, its dynamics and peculiar procedures—start from zero at the first or second year of academy. To better associate examples provided inside the classroom, and execute the initial roles onboard, the students need practice, usually coming with time after each training onboard. The use of a system like YPSim would certainly help to start building this basic mental model representation of the YP, producing results on both sides: classroom and onboard. Providing an intermediate level of experimentation media, YPSim would provide a chance to enhance the midshipman's representation of the YP world and stimulate the understanding of new topics in the classroom. This implies that less time is required to reach a desirable knowledge level to execute tasks onboard, either individually or as a team.

The relevance of prior experiences in a given environment is reinforced by the theory of *Perceptual Cycle*. Neisser (1976) explains that perception is a constructive process that connects an anticipatory schemata, the environment exploration, and information acquired by modifying the schema. First-year midshipmen have little knowledge to build a navigation schema in their minds. Assuming they have never been onboard the YP, it will be difficult to direct their attention to the correct visual information necessary to execute a task. If he/she plays the role of helmsman at the YP's bridge, his anticipatory schemata will direct his attention to the wheel, the only information known to correlate to his/her task. Information available from the compass, rudder angle indicator, and the outside world are probably neglected the first time. The novice helmsman does not understand that new cues are relevant until the experimentation—sometimes directed by external guidance from an instructor—changes his/her anticipatory schemata. Imagine now the OOD role, usually played by fourth-year midshipmen who are more experienced onboard. It takes time to construct a good OOD schema, one that drives the midshipman's attention to the proper set of cues

available at the YP’s bridge. YPSim represents a supplementary source of experimentation in a virtual environment presentation of the world to be explored.

C. CURRENT GAME-BASED SIMULATIONS

One of the first paragraphs of *Naval Shiphandler’s Guidebook* concerns the advancements of simulation in shiphandling training:

> The most important and valuable advancements in shiphandling training has been the introduction of shiphandling simulators. More than fifty simulators that meet international standards exist in the United States, and there are more than two hundred worldwide. In addition to these full-sized devices, there are countless simulators of lesser capability, as well as model boat basins, that offer valuable training opportunities for shiphandlers to learn and to improve proficiency. (Crenshaw, 1975)

According to Caird (1996), “visual simulation” is a 3D graphic image generated by computers to create a cognitive and physical interaction (Caird, 1996). This interaction is represented by the cognitive and physical real-time participation of the user in the environment (Salvatore, 2005). Bringing immersion, interaction and presence into a virtual environment that represents the working world where navigation and shiphandling take place, trainees can experiment and practice skills in a controlled physical situation. The simulation benefits for training are largely explored in previous works (*e.g.*, Norris, 1998; Grassi, 2000; Brannon and Villandre, 2002; McDonough and Strom, 2005; Salvatore, 2005; Ernst, 2006, Toledo, 2006; Rodrigues, 2010; Brown, 2010).

In the past, shiphandling simulators, often called *Bridge Simulators*, were associated with large hardware infrastructure capable of rendering 3D graphics and computing an acceptable physics model of the simulated ship. Today’s reality is different, and a commercial computer can run powerful simulations with the same characteristics of a Bridge Simulator decades ago (Salvatore, 2005). The Full Mission Bridge Simulators (FMBS), where trainees physically operate the equipment of a bridge as a team, are still in use and represent the biggest effort of industry in this arena. FMBS are expensive and demand a large

personal and material structure to operate, making a team training a big event in a sailor's day. A cheaper and easily accessible alternative—although for individual training only and less realistic—is the PC-based simulator. Bridge simulations running in conventional PCs—referred in this thesis as *serious games applications*—are a perfect solution for part task training, individual training, and learning basic skills. As described in Section A (Navigation and Shiphandling Instruction), midshipmen needs for a simulator are more towards the simplicity and accessibility of a game-based simulator rather than an FMBS. The level of complexity of the skills intended to train in this work does not require the use of an FMBS. This thesis focuses its attention on the use of a game-based approach to address issues concerning navigation and shiphandling instruction at naval academies.

Game-based systems are defined as any computer-supported real-time system that couples multiple sensory information in a organized way, providing a meaningful context for human action and collaboration (Sadagic, 2010). This definition also includes the following elements as characteristics of a game-based system:

- Content: representation of environment (typically 3D), actors and characters (one or many)
- Storyline: giving a meaning to all actions
- Roles: all actors and characters have them
- Tasks and goals: well-defined objectives
- Dynamics: set of rules, behaviors and interaction modalities
- Competition: levels, teams, scores

Zyda (2005), Salvatore (2005), and Ernst (2006) also include motivation and involvement as key factors that make game-based systems more attractive when training in situations where primary users are part of the "wired" generation. This meets the purposes of YPSim, primarily designed to attend young midshipmen ranging from 18 to 22 years old. By leveraging midshipmen

curiosity, each becomes personally motivated to learn, creating an attitude of intentional learning, resulting in the retaining of useful information (Ernst, 2006).

Experiences from games are capable of creating stories on top of some pedagogical content. Using stories as a tool, humans can gain understanding, solidify memory, and have an opportunity to tell their own version of any event (Beard, 2006). While playing a game, users can make connections between theory, concepts and ideas, and a story representing these abstractions in a meaningful and memorable way. These connections can help a future retrieval of information if associated with any significant event that happened during the story played in the game. Consider a situation where a midshipman executes a mooring task in a simulation game, where he is the OOD responsible for the ship maneuver. If, during the pier approach, the YP's speed was too high, thus resulting in a collision against the pier, this significant event can be registered as a story in a mental model representation that could be retrieved for later review and correction.

> Mental Simulation and Storybuilding Mental models can be used to mentally project into the future. We tell ourselves stories about the situation as it unfolds, and we mentally explore alternative, hypothetical futures. Whereas mental models provide a causal understanding of how situations came about and what they are in the present, mental simulation involves enacting series of events and pondering them as they lead to possible futures. Like mental models, mental simulation and story building are essential to sense making, problem detection, and decision making. (Crandall, 2006)

The accessibility of a game-based system, installed in a lab desktop or even in the trainee's own PC, represents a big advance in training. Imagine a midshipman after a class where the instructor presented some theory about mooring a ship. This task, in real life, requires a very complex set of skills from the OOD: wind and current perception, engines and rudder control, twisting and pivoting, mooring lines handling, etc. If the midshipman leaves the classroom with doubts about the "what ifs" scenarios in a mooring maneuver, the training aboard the YP will represent the only place to experiment and clarify his/her

knowledge. If an FMBS were available at the academy, his/her hypothesis about the maneuver could also be tested, but in a limited amount of time and depending on simulator availability. Under the supervision of instructors and other peers, this midshipman would typically be hesitant to make mistakes or try nonsense experiments with the virtual ship. However, these pressures and restrictions do not exist when using a game-based system running on the midshipman's PC. He/she could start answering questions and testing maneuvering hypothesis right after the class, without any delay or assistance.

The concept of using game-based ship simulations for training is increasing dramatically with the hardware improvements of PCs. The advance of multi-core CPU and GPU capabilities is opening a new dimension for more detailed physics models and high-resolution graphics running on a desktop or even a laptop. Three applications designed for seamanship simulations are relevant to the scope of this thesis and deserve special attention: SurfTacs, Fleetman Desktop, and Ship Simulator.

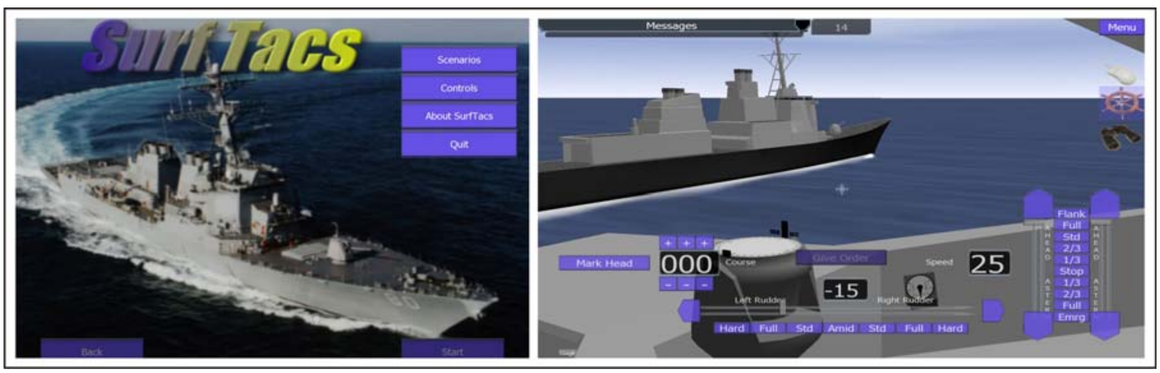


Figure 4. SurfTac's screenshots.

SurfTacs was a Master's Thesis project designed by Ernst (2006) at NPS. Adopting open source concepts described by Salvatore (2005), SurfTacs was developed using the Delta3D game engine as main library infrastructure. SurfTacs is a stand-alone Division Tactics Simulator (DIVTACS) intended to train tactical maneuvers aboard a virtual Arleigh-Burke Class Guided-Missile Destroyer (DDG) (SurfTacs, n.d.). SurfTacs represented a proof-of-concept

implementation of a game-based systems, using Delta3D game engine, to partial task train for OOD tactical skills. Salvatore (2005) and Ernst (2006) play an important role in the design of YPSim. YPSim inspiration comes from Salvatore's configuration and architecture for an open source simulation using Delta3D and Ernst's proof-of-concept implementation of SurfTacs. Despite the fact that YPSim is not a continuation of Ernst's work in SurfTacs, both share the same concept of a game-based system using Delta3D infrastructure. While SurfTacs is designed for tactical maneuvers aboard a DDG, YPSim focuses on navigation and shiphandling aboard the Naval Academies' YPs.

Figure 5. Fleetman Desktop's screenshots.

The second software is Fleetman Desktop, developed by DTM Global, a British company. The Fleetman family of bridge-training simulators is designed for team training types of scenarios. The Fleetman Desktop variant offers individualized training for OOD skills regarding navigation, radar operation, contact risk managing and UNREP. Fleetman Desktop is a proprietary software used by naval forces including the UK Royal Navy, Royal Navy Oman, the Royal New Zealand Navy, and Royal Saudi Naval Forces. Network capabilities offer options of missions in a distributed virtual environment, which expands the simulation's training capabilities and instructor's guidance (Fleetman Desktop, n. d.).

Figure 6. Ship Simulator's screenshots.

The last seamanship simulation of interest in this work is the Ship Simulator family. Ship Simulator is a commercial software developed by the VSTEP, a company from Holland. Ship Simulator is a PC game simulator usually associated with a maritime version of Microsoft's Flight Simulator. Ship Simulator Extremes is the revolutionary latest game in the best-selling Ship Simulator Series. Featuring a realistic ocean system, advanced dynamics and weather systems, Ship Simulator offers high-quality graphics representing missions where users can play the role of a captain in several types of ships (Ship Simulator, n. d.). Ship Simulator was not designed primarily as a training tool; rather, it was focused on the entertaining aspects of driving a ship under different scenarios. Despite its original purpose, the game offers fairly realistic conditions of a ship's dynamics and control operations, where users can practice numerous ship-driving skills.

D. CONCLUSION

This chapter reviewed basic topics required for a better understanding of the training issues and respective solutions explored in this thesis. First, the framework for navigation and shiphandling instruction adopted at USNA and BNA was analyzed. Assuming the similarities between frameworks at both USNA and BNA, and the generalization of the findings in this research, the BNA framework was selected as a reference model for investigation and design of YPSim. A

knowledge gap between the classroom and hands-on training instructions was identified, reducing the effectiveness of the YP as a training resource. Second, the cognitive process involved during training aboard the YP was explored. Referring to the theory of perceptual cycle (Neisser, 1976), the use of part-task simulation was viewed as a potential solution to address the knowledge gap issue. Exposing midshipmen to an extra exploratory step in the new environment represented by the YP can lead to a rapid change in their schema, leading to a better mental model relative to the tasks covered by the hands-on training. The third topic reviewed concerned the game-based systems simulators approach as being the best solution for the type of system needed for this training simulation solution. A brief analysis of three ship driving game-based systems (SurfTacs, Fleetman Desktop and Ship Simulator) was conducted. YPSim's design should be guided by SurfTacs' training concept and architecture, using Delta3D, adding the graphical appeal and game structure of the Ship Simulator family, a challenging task.

In order to raise the design requirements for YPSim, the next chapter presents a description of the missions executed by midshipmen during the hands-on training aboard BNA's YPs. From the set of missions enumerated, two of them will be selected for a cognitive task analysis study from the perspective of the OOD. The design requirements will be guided towards a maximization of the exploratory steps of the midshipman's perceptual cycle, presenting the maximum number of cues involved in their cognitive processing.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MISSION DESCRIPTION AND COGNITIVE TASK ANALYSIS

The process of designing a training system, such as the Yard Patrol Simulator (YPSim), necessarily involves the analysis of its training objectives. Providing answer to the following questions is key to achieve this goal:

- Who will be trained by the system?
- What do they need to train?
- In what level of detail this training will be given?

The first step is to define who will be the principal users of the system. It is important to provide relevant content that better suits the expected population of users, facilitating acceptance and usability. A training system designed for young midshipmen should not have the same interface as one designed for senior officers. Exploring the motivational factor that a 3D simulation game can bring to the training environment is important. Using the appropriate set of stories and symbols will trigger more enjoyment and satisfaction in that specific group of users.

The next step is to elaborate a brief description of each mission simulated and trained using YPSim. Navigation and shiphandling are two vast fields, and one could imagine an almost infinite combination of parameters and tasks when elaborating a mission to be trained. YPSim has a very specific application that is bridging the knowledge gap between classroom and hands-on training at sea. There is no need to design YPSim with celestial navigation training capabilities, for example, since this type of task is not performed aboard of the YPs. On the other hand, tasks such as Man Overboard (MOB) and Anchoring are essential and extensively trained there. The identification of simulated tasks represents an important step towards a good design of YPSim.

After understanding who will be the final user and what type of missions he/she will perform using YPSim, a complete analysis of each task performed inside these missions is necessary. Using a cognitive task analysis methodology,

the relevant details about how the midshipmen will accomplish his/her goal and the resources required to do that will be listed. Knowing that the system needs to be designed to provide MOB recovery types of missions is not enough information. Understanding how the user will observe the apparent wind information from the sensors, and what information will be retrieved to indicate the correct timing of a turning, for example, is also very important. The CTA study will gather all the necessary aspects and details of each mission, providing answers to the third question and defining the appropriate level of detail for the system.

A. MISSION DESCRIPTION

The YPSim's purpose is to simulate, in a 3D game environment, some of the basic missions performed during hands-on training exercises aboard naval academy training ships, as known as YPs. Among a whole set of missions usually performed aboard the YPs, the ones that have greater association with classroom instruction are:

- Tactical Maneuvering
- Underway Replenishment
- Mooring/Unmooring
- Man Overboard (MOB)
- Basic Navigation
- Anchoring

The above missions represent the most important set of tasks to simulate with YPSim. Other types of missions could be also explored, but they are considered as having minor effects on the system's design, and could be described as a future work.

To conduct the YP operation, each midshipman aboard is assigned to a specific watch, working as a team. Midshipmen, when assigned to a watch, are entrusted with the safe and proper operation of the YP (USNA, 1991.). This

thesis focuses on the following watches, and respective responsibilities, conducted by midshipmen at the YP's bridge:

- Officer of the Deck (OOD): has duties of supervision of the ship's operation and safe navigation. The OOD issues necessary orders to helm and main engines control to maneuver the ship
- Junior Officer of the Deck (JOOD): OOD's assistant
- Quartermaster of the Watch (QOW): assists the OOD in all navigation matters, maintaining the YP's navigational track and makes timely recommendations to the OOD concerning the YP's position
- Helmsman: stands watch at the YP's helm steering the ship
- Lee Helmsman: stands watch at the YP's engine order telegraph
- Radar Operator: stands watch at the navigation radar (BNA's YP only) reporting bearing and distance information from contacts or landmarks to OOD

Figure 7. BNA's YP bridge overview.

A brief description of the previously listed missions is given below, with respect of the tasks performed by the midshipmen on watch at the YP's bridge. The BNA YP's concept of operation was used in this description, assuming small—but not significant in the scope of this thesis—differences to the USNA YP.

1. Tactical Maneuvering

During this type of mission, the OOD is responsible for maneuvering the ship to a given position relative to another ship in a formation, designated as the guide ship. Any ship in a formation can be assigned as the guide. The OOD will give orders for engine and rudder to Helmsman and Lee Helmsman, respectively, in order to change the ship's course and speed to reach the correct station in a formation. Once on station, the OOD's responsibility becomes keeping the ship's position relative to the guide. An initial procedure preceding every maneuver is interpreting the tactical signal transmitted by the Officer in Tactical Command (OTC). The orders given by the OTC, telling all ships to maneuver and assume a given formation, will be transmitted using tactical signs. One of the skills trained aboard is how to translate the tactical signal code into meaningful context. The competent naval shiphandler must know the contents of these publications (the tactical signs codes) in detail, for when maneuvering in formation, there is rarely time to look up the answer (Barber, 2005). After transcribing the tactical signal into plain language, the OOD will be able to calculate his/her station to assume in the formation, course, speed and time to get there. Exercising the skills of interpreting tactical signals, calculating stations in a formation, and the course and time to maneuver are complex tasks with which a midshipmen will require time to become comfortable. The YPs are a great environment in which to conduct this kind of training, but the amount of midshipmen to be trained and the mistakes that usually happen make it an effective training for only a few students. Using a game simulation like YPSim, the basic skills could be explored individually and repeated until the midshipman reaches a good level of knowledge before going to the real YP experience.

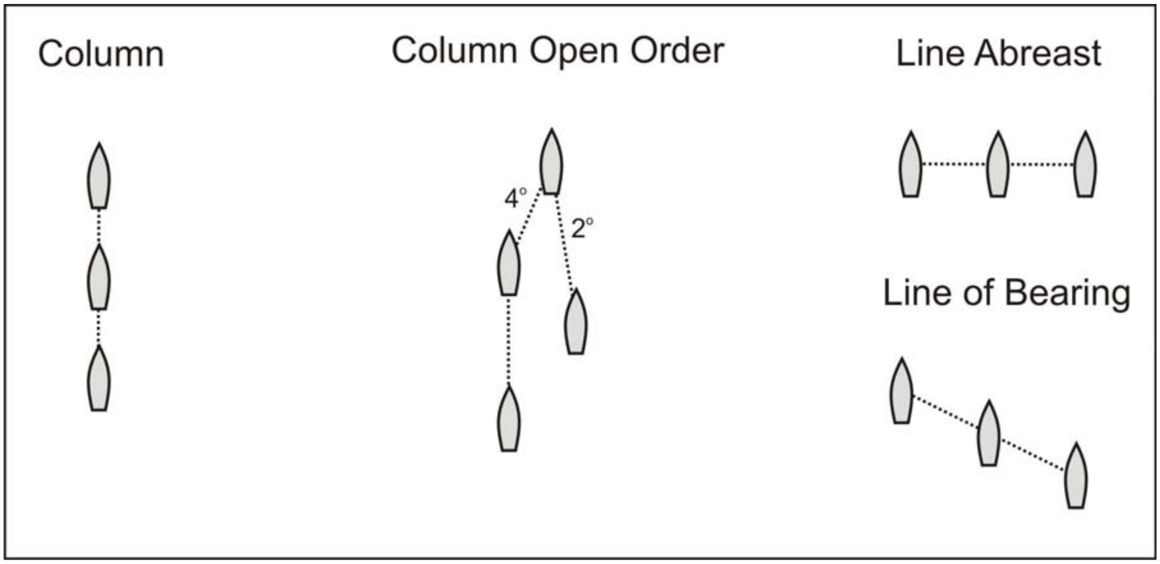


Figure 8. Typical ship formations used during simple tactical maneuvers exercises with YPs.

2. Underway Replenishment (UNREP)

UNREP consists of a maneuver where one ship transfers supplies to another, while steaming on parallel courses and linked up alongside (Barber, 2005). This is a high-risk maneuver in real life since two ships must be in close proximity for an extended time, another good reason to be trained in a simulation. According to Barber's (2005) "Naval Shiphandler's Guide," the UNREP follows an established pattern and can be summarized as below:

> The Officer in tactical command orders a course and speed (Romeo Corpen and speed) for the evolution. The delivering ship comes to the ordered course and speed, and signals on which side the receiving ship should approach. The entire operation is frequently conducted in radio silence, using flag hoists. The receiving ship moves into waiting station three hundred to five hundred yards behind the delivering ship, offset by the distance it will be while alongside, usually 120 to 200 feet (20 to 60 feet for YPs, according USNA (1991)), depending on the size of the receiving ship. When the delivering ship signals her readiness, the receiving ship increases speed and move alongside. She then holds the position alongside while the transferring rigs are passed and tensioned, the transfers are completed, and the transferring

rigs returned to the delivering ship. As the last line goes clear the receiving ship increases speed and clears out ahead, gradually increasing separation as it goes. (Barber, 2005)

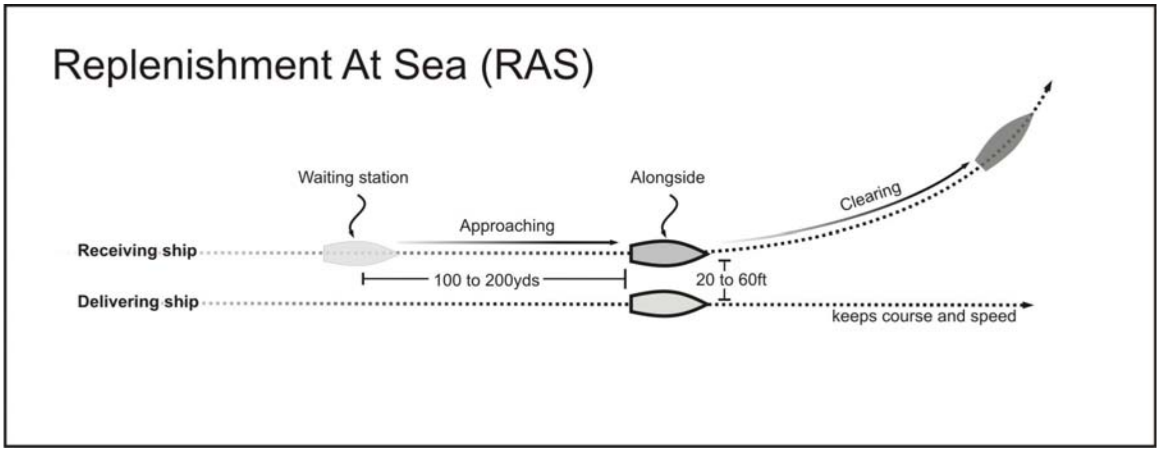


Figure 9. Underway replenishment diagram. Due to size proportions, UNREP distances for YPs are significantly smaller than the used in the fleet.

The UNREP description provided by Barber apparently seems to be easy and simple and in fact, it is not. His words represent a simplification and a model of something much more complex when executed. Having a good feeling about the ship's dynamics and effects of controlling actions (rudder and engines) is crucial to a good performance during UNREP. It takes time to gather these types of skills, and confidence comes with trial-and-error experiences over much training. Usually, one maneuver like this will consume between 40 and 60 minutes in a expedite mode aboard a YP, and 15 minutes more to another execution. Considering that each daily YP training section is approximately four hours long, at most, three midshipmen could be trained as OOD for this mission aboard the YP per day. Using a game simulation, one single midshipman could experiment more than five UNREP maneuvers in the same time length.

3. Mooring/Unmooring

Defined by Grassi (2000) as Pier Side Handling, this type of maneuver is overviewed as:

> One of the most basic, yet extremely critical, evolutions performed by a OOD. It is also one of the most rewarding evolutions. For if a OOD can smartly and safely accomplish a pier side evolution, it demonstrates to his peers how good of a ship driver he or she really is. Successfully accomplishing this evolution, however, takes planning, advance preparation, training, and the teamwork of everyone involved. (Dodge, 1981)

Mooring a ship simply means the evolution of bringing the ship alongside a pier using rudder, engine and mooring lines. Sometimes, external factors will be available to maneuver, such as harbor tugs. Often, environmental conditions such as wind and sea currents will be present, and the officer needs to know how to play with these factors in order to compensate or be benefitted by them. Unmooring is the opposite; using the same resources (rudder, engines, mooring lines, tugs, wind and current), the OOD will get underway from a pier. In a regular situation, a fully operational YP moors and unmoors without the help of a harbor tug. The crucial element of this task is understanding how to use all the available forces in order to move the ship to the desired position. During mooring/unmooring, ships behave differently than underway due to the lines connecting with the pier. This factor will drastically affect a ship's pivoting point and ability to move. In addition, at very low speeds, dramatic changes happen in the way rudder forces, wind and current move the ship. The feeling of this dynamic situation is hard to understand in a classroom instruction, and the only chance to practice this concept is aboard. Each ship has its own characteristics; even ships of the same class act differently in pier-side shiphandling evolutions. Basic concepts, on the other hand, can be easily explored using a game simulation with a simplistic physics model, as is the case of YPSim.

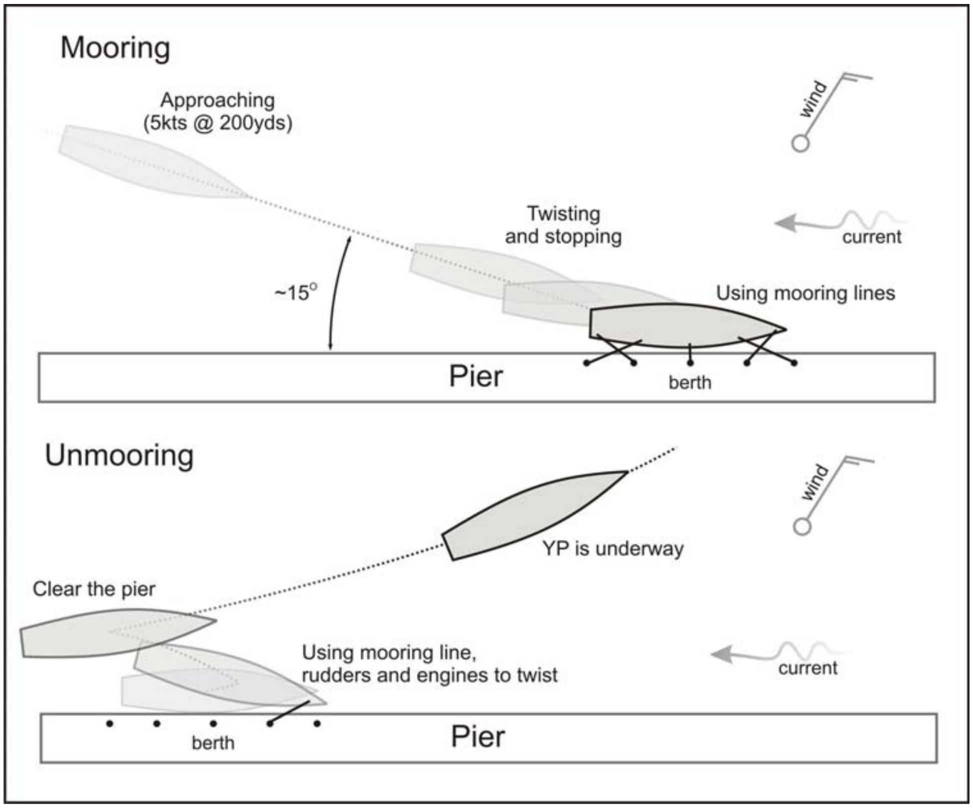


Figure 10. A basic visualization of the mooring and unmooring missions.

4. Man Overboard (MOB)

There are many standardized maneuvers to accomplish the task of recovering a person from the water. Each one of them has its pros and cons but, in general, midshipmen practice two of them onboard the YPs: Williamson; and Anderson.

> The classic man overboard recovery maneuver is the Williamson turn, developed during World War II by Cdr. John A. Williamson. The goal is to reverse course in a way that heads a vessel back along its original track. When properly executed, the maneuver will place the ship in her own wake on a reciprocal course. The Williamson turn can be useful for recovery of a person whose position is known. It is almost imperative if the person's position is not known. The classic Williamson turn starts with full rudder to the side on which the individual went over, if known. Continue with full rudder until the ship's heading is 60 degrees past the original

> heading, then shift to opposite full rudder. The momentum of the turn will normally carry the ship's heading to about 90 degrees from the initial heading, at which point the bow starts to swing in the opposite direction. Continue the turn with full rudder to steady up on the reciprocal of the original track about one turning diameter away from the point at which the turn started. (Barber, 2005)

The Anderson turn is simpler, and it is used when the person is in sight, under good conditions (Barber, 2005). It consists of a full rudder continuous turn with a full engine ahead at the beginning, and decreasing speed from the half-way to the end.

> Until you have this maneuver accurately calibrated it is better to take off speed a little early than to risk overrunning the person in the water. The back bell (engines backwards) can be adjusted either up or down to bring the ship to a stop at the chosen pickup point. This will normally be with the person in the water a few yards to leeward of the ship's bow. The ship will drift downwind at a greater rate than the person in the water. Care is needed to keep the person being recovered forward of the ship's intakes and screws. (Barber, 2005)

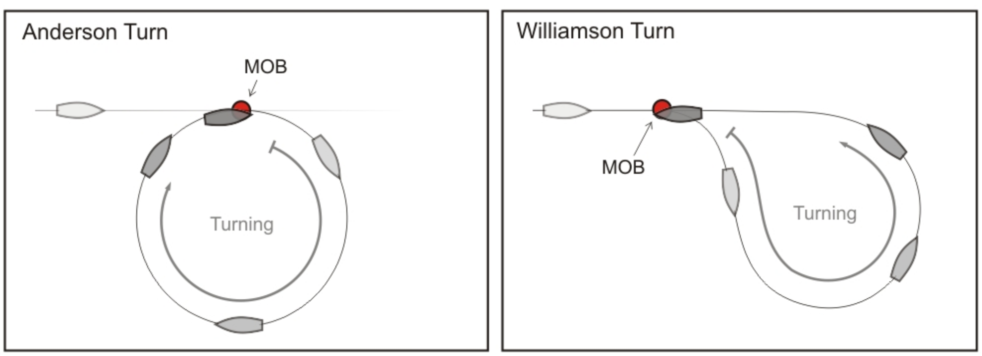


Figure 11. An illustration of Anderson and Williamson Man Overboard maneuvers.

The OOD also executes a series of non-maneuverable procedures during a MOB mission, such as giving orders to hoist Oscar flag, short blasts of the ship's whistle six times, checking the plotting of MOB position on the nautical chart and GPS, and giving orders to throw pyrotechnics to signal the individual's

position, among others. These procedures are also evaluated aboard in training missions. YPSim could easily simulate both maneuver and administrative procedures for the MOB mission.

5. Basic Navigation

Missions composed of moving, in a safe and efficient way, the ship from point A to point B are defined as Basic Navigation. It is during these types of missions that navigation teams get team training on many navigational topics—from classroom to reality. The calculations and procedures that determine a ship's position are not part of the OOD role, but he/she still has a lot to learn and train for during Basic Navigation. The OOD will be responsible to evaluate the information provided by the navigation team, keep the ship safe from other contacts and hazards, and have a complete picture of the outside world in his/her mind. A navigation route is usually planned in a way that navigation aids, such as buoys, lighthouses, and landmarks can be used as references for turning points along the way. Another aspect of interest explored by missions of this sort is the alignment of conspicuous points on the horizon, such as towers and lighthouses, which give the OOD an exact line of position on the nautical chart. During navigation, the OOD can explore the use of the radar, experimenting with controls and settings while obtaining bearing and range of other contacts or hazards. The familiarity with all these basic concepts and equipment is something that takes some time to gather during the hands-on training exercises, but could be easily simulated in a virtual environment of a 3D game such as YPSim.

6. Anchoring

Anchoring is a precision task that involves a basic navigation mission, ending in an anchorage position, followed by a set of standardized procedures to set the anchor. The objective is to follow a pre-defined route that ends up at the anchoring position, usually with a few waypoints before getting there. The OOD

is responsible for (1) giving the appropriate orders of rudder and engine that will keep the ship close to the planned track, (2) execute turns at the waypoints, (3) controlling effects caused by wind and sea current, and (4) setting anchor at the right position. If a desired ETA (Estimated Time of Arrival) is required, OOD will be responsible for controlling the YP's speed and set anchor at a specific time. The navigation team is, again, responsible for determining the ship's position and providing suggestions on the course and speed to the OOD, who will judge whether to accept it or not. Before reaching the anchorage position, OOD releases some important information to the Boatswain at the forecastle, providing depth at anchorage, scope of chain to pay and the nature of the seabed. During the anchoring, the Boatswain stays at the forecastle with his team operating the anchor gear and reporting anchor status to the OOD. On reaching the anchoring position, the OOD stops the ship, drops the anchor and moves the ship in reverse until it is secured and will no longer move. Knowing that an anchor is properly set on the seabed and securely holding the ship is fundamental to assess whether the maneuver is over. Instruments and environment will provide cues for the OOD to evaluate as to whether the ship is anchored or not, such as relative movements of fixed land points and water flow alongside the ship.

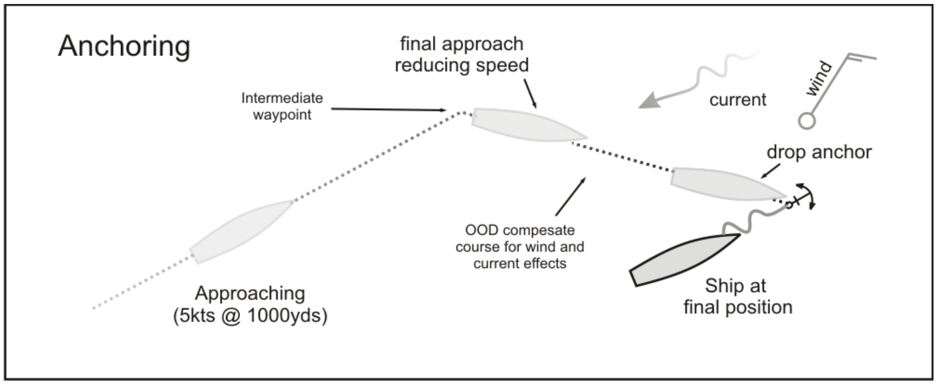


Figure 12. A simplified representation of the Anchoring maneuver.

B. COGNITIVE TASK ANALYSIS (CTA)

According to Crandall (2006), CTA is a family of methods used to study and describe reasoning and knowledge, applied by trainers and instructional designers to identify processes to be trained and how to best train them. CTA purpose can also be represented by modeling the actions, knowledge and cognitive process used by an individual performing a task (Jonassen, 1999).

Although CTA is not the main research focus of this thesis, it represents an important tool when designing requirements for YPSim. Identifying the most important tasks to simulate, the critical cues necessary to provide, the actions required, and the set of skills expected to be learned while playing the game is fundamental. A good example is during a Man Overboard mission, when the OOD uses the ship's wake as a very important visual cue to identify how fast the ship is turning and assess whether any controlling action is required. If the ship's wake is not presented in the simulation, this cue will be omitted and the instructional potential of the game therefore affected. This type of understanding about the user playing his/her role in the system is absolutely essential to a good system design. The use of this methodology leads to the establishment of learning objectives and performance metrics used in a feedback and scoring component of the system. It is not only important to enumerate the system's capabilities but also making clear what it is not intended to train. Trying to use a game simulator to train complex missions or using it as a realistic physics model of the YP could lead to a potential negative training transfer.

For the scope of this thesis, only two missions among the six previously described will be analyzed in detail: (1) Anchoring, and (2) Man Overboard. These two missions were selected because of their generalizable set of requirements, hoping to provide a solid framework for future development of the remaining four missions.

This CTA is focused only in tasks performed by midshipmen playing the OOD role aboard the YP. The OOD is the duty of higher responsibility at the YP's

bridge, and is usually carried out by a senior midshipman. YPSim is designed for senior midshipmen to serve as the primary users playing the OOD role in the simulation. Adopting a three-step approach for studying the cognitive process of the two selected missions, Man Overboard and Anchoring, the CTA in this work is presented in the following sequence:

- Conduct a Hierarchical Task Analysis (HTA)
- Elaborate a Critical Cues Inventory (CCI)
- Identify performance metrics

The next sections describe the steps above, using Observations and Document Analysis techniques for collection of preliminary knowledge (Clark, 2008). The author used an adaptation of standard procedures listed on Barber (2005), Bowditch (2002), Crenshaw (1975), Hooyer (2004) and Miguens (n.d.), all reference manuals for shiphandling at USNA or BNA, as source documentation. All procedures were adapted to the BNA YP's operation, which can differ in some small aspects from the USNA reality. The author's observations of tasks conducted in 2008 aboard a BNA's YP were also used as a source of preliminary knowledge. Initial versions of HTA and CCI were constructed from the list of procedures and observations and submitted for review to three former YP instructors at the BNA. The results presented here in this thesis are previously reviewed versions.

1. Hierarchical Task Analysis (HTA)

Details and hierarchy of the tasks performed by the OOD, during the Man Overboard and Anchoring exercises aboard the YPs, are captured in this section. To offer a better understanding of the tasks hierarchy and chronological sequence of actions, we build a summary diagram. The detailed description of each task node in the hierarchical diagram is listed in a table format, presented in Appendices H and I. The task's details contain valuable information about some basic cognitive processing, sensorial perception and interactions performed by the OOD.

a. HTA for Man Overboard (Anderson Turn) Task

The tasks conducted by the OOD on a Man Overboard mission aboard the YP, as described in Section A.4 of this chapter, are summarized in Figure 13. Detailed information of each task are presented in Table 13 (Appendix H).

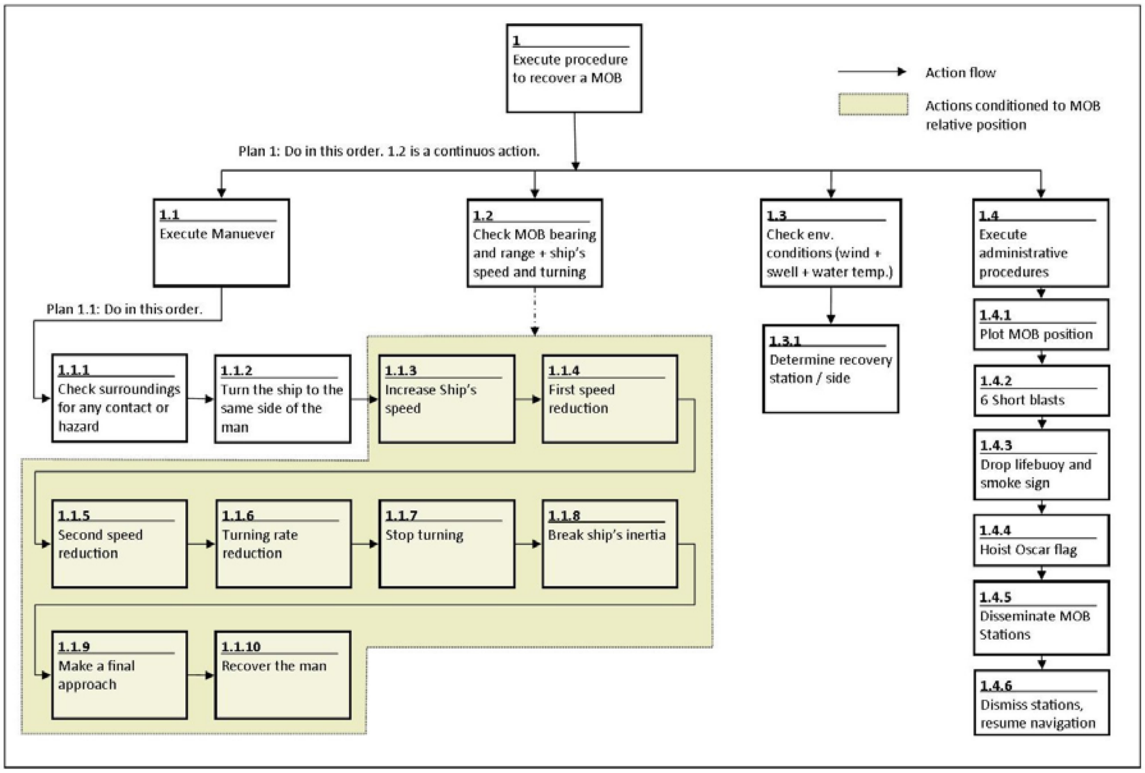


Figure 13. Hierarchical diagram for the Man Overboard task.

b. HTA for Anchoring Task

The tasks conducted by the OOD on a Anchoring mission aboard the YP, as described in Section A.5 of this chapter, are summarized in Figure 14. Detailed information of each task are presented in Table 14 (Appendix I).

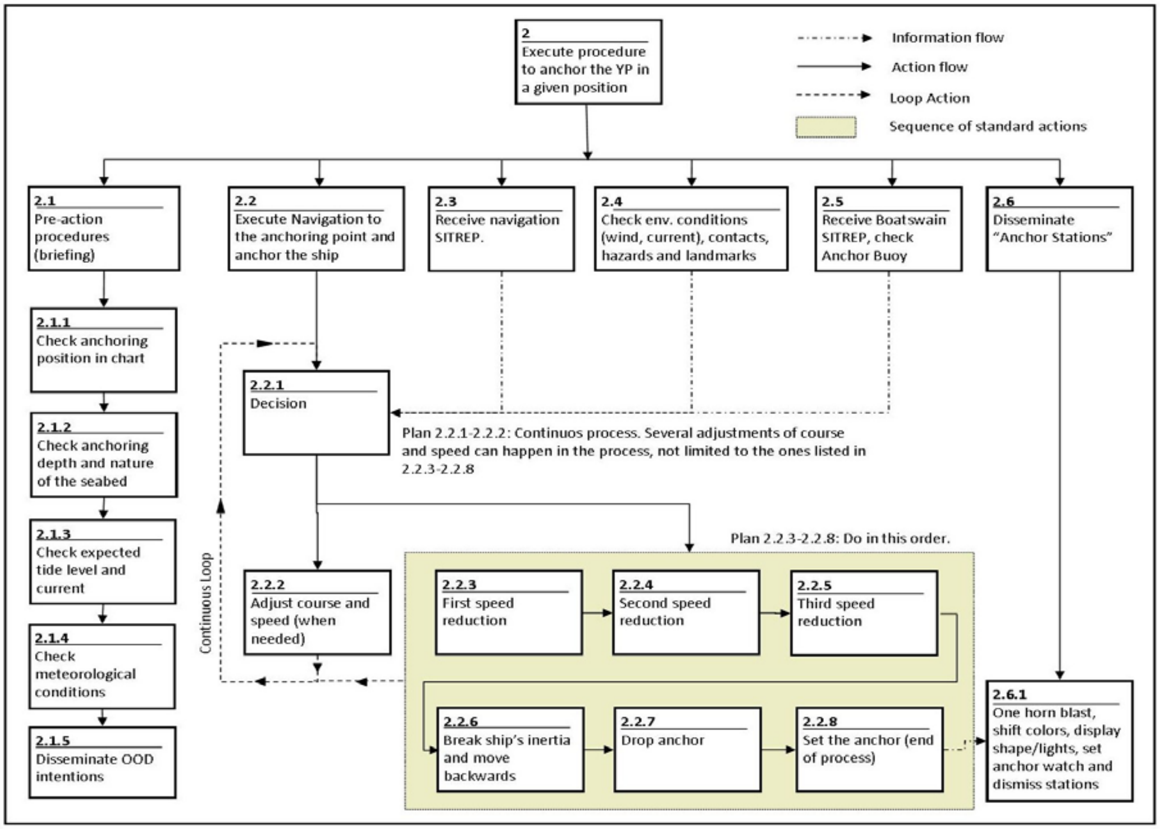


Figure 14. Hierarchical diagram for the Anchoring task.

2. Critical Cue Inventory (CCI)

Defined as a collection of informational and perceptual cues that are present when executing a given protocol (Klein, 1998), the CCI is an important tool for the design of YPSim. Once the identification of the CCI is done, the designer will have a complete list of the information required by the user during the simulation. Grassi (2000) successfully used this technique in his study for a "Virtual Environment Pier Side Shiphandling Simulator." This thesis adopts a CCI's format similar to the one used by Grassi (2000), with perceptual cues listed on the lefthand side and their descriptions on the righthand side. Each CCI table refers to a specific decision or action step on the hierarchical diagram for both MOB and Anchoring tasks. At each of these steps, perceptual cues and information were analyzed and described, showing how and where OOD acquire

them, and how they are used in the maneuver process. The CCIs were divided according to each specific mission explored in this work, i.e., MOB and Anchoring.

a. CCI for Man Overboard Task

Table 1. Critical Cue Inventory for checking surroundings before turning.

Critical Cue Inventory for:	
Step: 1.1.1	Check Surroundings for Contacts or Hazards
Cue	**Description**
Visual at the bridge wing	OOD goes to the bridge wing of the side of the intended turn (port or starboard) and visually observes the surroundings at close distance. Any contact, such as ships or boats, at that side and at close range should be visible, assuming a good visibility condition for this type of maneuver (Anderson Turn). OOD also looks for close distance navigation hazards such as buoys or land obstructions that could interfere while turning.

Table 2. Critical Cue Inventory for checking MOB's bearing and range.

Critical Cue Inventory for:	
Step: 1.2	Check MOB bearing and range
Cue	**Description**
Visual at the bridge wing	OOD stays at the bridge wing (of the same side of the turn) during the maneuver, observing the MOB. Binoculars may be used to facilitate sighting. Smoke sign and lifebuoy will help to spot MOB's position. Bearings can be read on the gyro repeater located at both bridge wings (port/stbd.). MOB distance is more difficult to precise, however the relative notion of how fast the ship is getting close to the MOB is key. The rate of change of bearing marks is also important to guide OOD during the "final approach" step of the maneuver. MOB's rate of change on bearing and range represent the system feedback for OOD's changes on course and engines RPM.
Quartermaster of the watch report	Quartermaster of the watch verbally reports MOB bearing and range from GPS. OOD can dismiss this information if MOB is sighted, avoiding excess of noise during the maneuver.

Table 3. Critical Cue Inventory for checking ship's status.

Critical Cue Inventory for:	
Step: 1.2	Check Ship's Status
Cue	**Description**
Sound of engines	Used by OOD to audibly detect engine changing speed. At the bridge wing, this sound is noticeable and provides a good feedback to OOD if his/her engine orders were executed.
Gyro Repeater	Used by OOD to check the ship's heading, MOB's bearing and rate of swing of the ship. They are located at each bridge wing.
Engine Control Levers	Used by OOD to visually check the RPM status of both engines (port/stbd.). The engine control is located inside the bridge and operated by the Lee-helmsman. From each bridge wing, the OOD can quickly check the levers position of both engines, providing an important cue of engine status without asking Lee-helmsman. This resource is key since OOD can easily forget his/her last engine orders during the maneuver.
Rate of swing of ship's bow	Used by OOD to visually check the ship's response to his/her rudder orders. A good feedback to check if the rudder order was accomplished as desired. This cue is also key to OOD's evaluation for following changes in rudder.
Ship's wake	Used by OOD to visually detect engine changing speed. At the bridge wing, OOD can check ship's wake after giving any engine order and observe for any change as a feedback.
Relative motion of MOB	Both bearing and range of MOB are used by OOD to evaluate ship's movement towards the MOB itself. The MOB's relative motion guides OOD at the final approach working as a control point. Every maneuver decision made by OOD relies on this cue.
Helmsman and Lee-helmsman Acknowledgement	Used by OOD to check if engine and rudder orders were acknowledge. After every rudder or engine order, Helmsman and Lee-helmsman, respectively, verbally respond in a standard manner.

Table 4. Critical Cue Inventory for checking environmental conditions.

Critical Cue Inventory for:	
Step: 1.3	Check Environmental Conditions
Cue	**Description**
State of water	Used by OOD to estimate the direction and magnitude of the wind. The OOD is looking at the white caps or ripples in the water created by the wind (Grassi, 2000). The state of water will determine the amount of variation on course and speed ship will present at the "final approach" step. With calm seas, this variation caused by swell is little, increasing as the sea state gets higher.
Wind bird	Used by OOD to measure the magnitude and direction of relative wind. At the bridge wing, OOD has a good sight of the wind bird, located at the ship's mast. This cue is one of the most important to determine which side (port/stbd.) will be used to recover the MOB (should be placed at leeward).
Smoke sign	Used by OOD to estimate the true magnitude and direction of wind.
Pennants/Flags	Used by OOD to estimate the relative magnitude and direction of wind.
Water temperature table	Used by Quartermaster of the watch to retrieve the water temperature and inform OOD. OOD uses the water temperature to estimate the survival time of MOB.

Table 5. Critical Cue Inventory for administrative procedures.

Critical Cue Inventory for:	
Step: 1.4	Administrative Procedures (feedback of the actions taken)
Cue	**Description**
MOB symbol	Marked on the nautical chart, and/or GPS plotter, to indicate MOB's geographic position. Used by Quartermaster of the Watch to inform MOB's bearing and range to OOD. OOD can also obtain this information by directly watching MOB's position on chart/GPS.
Six Short Blasts	The sound of the ship's whistle is used to provide an auditory communication with ships in the vicinity. The six short blasts is a standard signal that indicates an emergency, in this case the MOB. This cue indicates the accomplishment of OOD's order (step 1.4.2).
Smoke Sign	OOD sees it floating in a position assumed to be close to the MOB, indicating that his order (step 1.4.3) was accomplished. The smoke sign is more helpful than lifebuoy to MOB's sighting, and will provide an extra visual cues for wind direction.
OSCAR Flag	Used to visually communicate this emergency (MOB) to other ships at visual range. When hoisted at the YP's mast, indicates that OOD's order (step 1.4.4) was accomplished. The OSCAR flag can be used as a secondary means of assessing relative wind speed and direction (the primary source in this case is the wind bird).

b. CCI for Anchoring Task

Table 6. Critical Cue Inventory for administrative procedures.

Critical Cue Inventory for:	
Step: 2.1	Pre-action procedures (briefing)
Cue	**Description**
Nautical Chart	Used by OOD to create a mental picture of the maneuver, including the entire navigation route up to the anchorage, possible hazards, expected landmarks and anchorage features. Important anchorage features are depth, nature of seabed and shelter level from winds and strong currents.
Tide Table	Used by OOD to estimate the tide level and current at the anchorage. The tide level is used to calculate the appropriate scope of chain while the current is one of the important forces acting on the ship during the maneuver.
Weather Forecast	Used by OOD to estimate the meteorological conditions at the anchorage. Wind and swell, are important factors in mental picture of the maneuver the OOD has to elaborate, since they could affect ship's behavior especially at low speed. The meteorological conditions at the time of the maneuver are relevant however, the forecast of these factors are also important to set the best anchorage and the correct scope of chain. When a bad weather is forecasted, very often the wind will come from a different direction and stronger intensity. This predicted change on the wind would lead to an sheltered anchorage that is different than the current situation solution.

Table 7. Critical Cue Inventory for Navigation.

Critical Cue Inventory for:	
Step: 2.2	Navigation to Anchorage
Cue	**Description**
Navigation Report	Standardized set of information regarding ship's navigation that is periodically (usually every three minutes) reported to OOD. Navigator is responsible for providing such report and verbally transmits to OOD. Used to create a mental picture of the current status of the ship's position in relation to navigation route. The reports contains information such as bearing, range and time to the next waypoint, offset to the route, suggestion of course and speed to keep ship on track, course after next waypoint, depth registered at the fathometer, range to the anchorage and calculated current affecting ship.
Visual of Surface Contacts	During every navigation OOD must keep track of potential hazards that could affect ship's safety while underway. One of the most important category of hazards are surface contacts navigating or not close to ship's route. OOD frequently scans the horizon for contacts and keep a mental picture of the ones that can affect his decisions about course and speed. Binoculars are an important artifact to improve OOD's situational awareness and evaluate contacts relative movement.
Visual of Landmarks	Used to reinforce OOD's mental picture of the nautical chart representation of the environment. Landmarks are used as references during the navigation and, more especially, when approaching the anchorage as a head mark.
Radar	Used to determine range and bearing of landmarks used as navigation references. Often used to estimate the relative movement of surface contacts, as well.
Anemometer	Used to evaluate direction and magnitude of relative wind that could potentially affect ship's Course Over Ground (COG). The anemometer panel is located inside the navigation bridge.
Fathometer	Used to evaluate current depth at ship's position. Depth is periodically provided by Navigator in his/her report, however this information is always available by OOD's request.
Gyro Repeater	Used to obtain bearings of landmarks, surface contacts and check clearance in a given direction. There is one gyro repeater located in each bridge wing.
GPS	Used to obtain ship's speed and course over the ground. When navigation route is set, can be used to retrieve bearing, distance, track offset and ETA to the waypoints on the route. The GPS is located inside the bridge and OOD is not always allowed to use it.
Speedometer	Used to read the ship's speed over the water. Navigator uses this information in his current calculation. OOD can use it all the times to check speed and evaluate his current engine orders.

Table 8. Critical Cue Inventory for Drop and Set Anchor.

Critical Cue Inventory for:	
Steps: 2.2.7 & 2.2.8	Drop and Set Anchor
Cue	**Description**
Boatswain Report	The Boatswain stays at the forecastle observing the anchor and chain at close distance after dropping it. Information about the anchor conditions (holding or dragging, how the anchor is tending - "9 o'clock," "12 o'clock," number of shots of chain, etc.), and what sort of tension is on the anchor chain will be vital to OOD properly set the anchor (USNA, 1991).
Visual of Landmarks	At very close distance to the anchorage, OOD needs some constant reference to assess ship's position. Navigator can only provide reports in some time steps that can be too long at the final approach. Knowing that some landmark or navigation aid should be at a given bearing at the anchorage can represent an important reference to OOD assess ship's position relative to his/her goal. Observations of two objects abeam and aligned at different ranges can provide OOD a reliable source for ship's movement when stopping for dropping the anchor.
Radar	At the final approach can be used by OOD to measure distance to landmarks ahead the ship. If the distance to this landmark is known at the anchorage and ship is on track, then OOD will have a reliable distance to his/her goal.
Anchor Buoy	A small buoy is attached to the anchor with a cable length proportional to the anchorage depth. This buoy is a valuable reference for OOD's mental picture of the underwater gear (anchor and chain). The OOD needs to pull the chain, usually moving the ship astern at low speed, in order to properly setting the anchor. Having a constant visualization of anchor's position will lead to a better maneuver.
Other ship's at anchor in the vicinity	Used to estimate the resulting forces of wind and current at the anchorage area. Ships, boats and other floating objects at anchor tend to align to the strongest force being applied (wind or current). This information is key to OOD predict the best direction to pull the anchor and easily set it.

C. PERFORMANCE METRICS

Instructors are constantly evaluating midshipmen performance, while training aboard the YP. The OOD midshipmen is the most important duty at the YP's bridge, and therefore the most demanding regarding performance evaluation. There are many aspects of a midshipman's action, while playing the

role of OOD, which require the instructor's attention. From leadership to safety procedures, the OOD's actions are always subject to evaluation. This work is focused on the OOD's shiphandling skills only; Some performance metrics for each one of the studied missions, Man Overboard and Anchoring, is described below. A broad view of how midshipmen are evaluated as the OOD aboard the Brazilian Naval Academy's YP will help the design of the system, concerning performance measurement and instructional feedback. These metrics will vary from one naval academy to another, and sometimes between YPs of the same institution, depending on instructor's style.

1. Evaluating OOD's Performance: Anchoring

Some generic performance metrics for OOD executing an Anchorage mission are described in Table 9.

Table 9. OOD's performance evaluation for Anchoring mission.

Indicator	Description	Metric
Track Offset	Overall evaluation of ship's offset from the navigation route up to the anchorage.	Distance (yards)
ETA Offset	Evaluating whether or not the ship arrives early or late to each waypoint.	Time (seconds)
Frequency of Changes	Evaluating how frequently OOD issued rudder or engine orders. An experienced OOD usually issue fewer commands than a novice. That's not always true since fewer commands could be a product of a poor decision process.	Counts per mission
Time to Set Anchor	Time between dropping and properly setting the anchor.	Time (seconds)
Anchorage Offset	Distance between the position anchor was set and the intended Anchorage.	Distance (yards)
Selection of Scope of Chain	Evaluating how well OOD selected the appropriate scope of chain to use. This value is a function of anchorage depth, tide level, nature of seabed, wind and current.	Length (shots of chain, one shot = 90 feet)
Evaluation of	Evaluating how well OOD evaluated the anchorage	Subjective

Indicator	Description	Metric
Anchorage Shelter	regarding its shelter level to expected wind, swell and current.	
Safety	Evaluating how well OOD's actions led to a safe navigation. Can be inversely proportional to the number of times safety rules were broken.	Subjective
Speed	Evaluating how well OOD followed the speed reductions schedule along the route.	Subjective
Engine Overload	Evaluating the number of times OOD put engines under overload conditions.	Counts per mission
Information Flow	Evaluating how well OOD communicated with Boatswain (reporting range to anchorage), Navigator, Helmsman and Lee-helmsman.	Subjective
Administrative Procedures	Evaluate if OOD accomplish a series of administrative procedures listed on Step 2.6.1 of Table 2.	Yes/No

2. Evaluating OOD's Performance: Man Overboard (MOB) Using Anderson Turn

Some generic performance metrics for OOD executing a MOB mission are described in Table 10.

Table 10. OOD's performance evaluation for MOB mission (Anderson turn).

Indicator	Description	Metric
Frequency of Changes	Evaluating how frequently OOD issued rudder or engine orders. An experienced OOD usually issue fewer commands than a novice. That's not always true since fewer commands could be a product of a poor decision process.	Counts per mission
Time to Recovery	Time between MOB is reported and recovery.	Time (seconds)
Recovery Distance	Evaluating how close to the recovery station the MOB was positioned at the end of maneuver.	Distance (yards)
Recovery Side	Evaluating OOD's decision about the recovery side (port/stbd.). The MOB should be placed at leeward at the end of maneuver.	Yes/No
Safety	Evaluating how well OOD's actions led to a safe recovery, avoiding placing MOB at risk.	Subjective
Initial Turn	Evaluating if OOD was able to move the stern away	Yes/No

Indicator	Description	Metric
	from MOB during the initial turn.	
Engine and Rudder	Evaluating how well OOD followed the procedures to Anderson turn, regarding rudder and engines orders.	Subjective
Engine Overload	Evaluating the number of times OOD put engines under overload conditions.	Counts per mission
Information Flow	Evaluating how well OOD communicated with Boatswain, Quartermaster of the watch, Helmsman and Lee-helmsman.	Subjective
Administrative Procedures	Evaluate if OOD accomplish a series of administrative procedures listed on Step 1.4 of Table 1.	Yes/No

IV. REQUIREMENTS ANALYSIS

A. OVERVIEW

The Requirement Analysis process is a collection of customer needs and objectives in the context of intended use, environment, and systems characteristics that will determine requirements for system functions (UAD, 2001). Using the Requirement Analysis process prior to the design phase in this research resulted in a better understanding of YPSim as a training system, its functions and constraints.

The framework adopted in this chapter is similar to that used by McDonough and Strom (McDonough, 2005) and Brannon and Villandre (Brannon, 2002) in their designs of Forward Observer PC Simulator (FOPCSIM) 1 and 2, respectively.

1. Purpose

YPSim is a game-based simulator primarily designed to provide a virtual environment representation of navigation and shiphandling tasks performed by the Officer of the Deck (OOD) aboard a naval academy's Yard Patrol craft (YP). The knowledge gap between classroom instruction and hands-on training aboard the YPs is expected to be reduced by the use of this game simulator, improving training outcome at sea.

As secondary applications, YPSim can be used as part task simulator for navigation and shiphandling courses in naval academies, as an instructional resource for classroom purposes, and as a motivational tool for naval academy students.

2. Users Demographics

The primary user population of YPSim is represented by naval academy senior midshipmen, male or female, ranging in age from 20 to 25 years, with previous basic classroom instruction in navigation and shiphandling.

YPSim's secondary users are naval academy midshipmen from other grades, male or female, ranging in age from 17 to 24. Naval academy instructors of navigation and shiphandling courses, with profound knowledge of navigation and shiphandling, ranging in age from 30 to 65, are also considered secondary users of the system.

3. User Environment

YPSim will be flexible enough to use as stand-alone software in a laptop computer, or configured using multiscreen and network capabilities in a lab environment for instruction. Naval academy midshipmen can use YPSim without an instructor's supervision, referring to the simulator's scoring system for a performance assessment. In a lab configuration, YPSim can be set to use a flight yoke type of controller, allowing three midshipmen to play different roles (OOD, Helmsman and Lee Helmsman) in the simulation, under an instructor's guidance. For classroom use, the instructor could execute YPSim using a wall projector in order to demonstrate basic concepts of the six missions described in Chapter III.

4. Technology and Dependencies

YPSim is a low-cost game simulation developed in C++, using Delta3D, an open source game and simulation engine maintained at the Naval Postgraduate School in the Modeling and Simulation (MOVES) Institute. The following Delta3D's external dependencies are required to run YPSim:

- OpenGL, for 3D rendering
- OpenSceneGraph (OSG), for scene graph organization
- OpenAL, for audio

- Open Dynamics Engine (ODE), for physics
- Crazy Eddie's Graphical User Interface (CEGUI), for Graphical User Interface (GUI)
- Game Networking Engine (GNE), for networking

B. SUMMARY OF CAPABILITIES AND LIMITATIONS

1. Capabilities

A list of desired capabilities and its correspondent benefits to YPSim are summarized in Table 11.

Table 11. Summary of Capabilities of YPSim.

Supporting Feature	Benefit
Use of environmental conditions on the scenario settings	Increase immersion and training scenarios
YP's physical model with good realism and real time performance	Increase immersion and training
3D Model of the YP's area of operation	Increase immersion
Multiple scenarios, at least one for each mission described on Chapter III.	Cover the full range of training scenarios aboard the YP
Multiple screens setup option	Provide flexibility for different configuration environments
Multiple input devices	Provide flexibility for different configuration environments
Multiple views option	Increase training visualization
Ocean model	Increase immersion and refine YP's physical model

Supporting Feature	Benefit
Performance Evaluation	Provide user feedback
Network	Increase training options for some missions that require interaction between users
YP’s instruments and controls simulation	Increase immersion and cues to task performing
Intuitive Graphical User Interface (GUI)	Increase usability
After Action Review (AAR)	Increase training feedback
Scenarios with navigational aids and other ships	Increase realism, immersion and training
Artificial Intelligence (AI) for scenario ships	Increase scenario realism
Mooring lines simulation	Provide mooring/unmooring training and increase immersion.
Anchor and chain simulation	Provide anchoring training and increase immersion.
Radar simulation	Provide radar familiarization, range and bearing source of information and increase immersion.

2. Limitations

YPSim is not designed to provide the following capabilities as a simulation training system:

- Full mission training to YP’s bridge team
- Extremely accurate and realistic physical model of the YP
- Intelligent Tutoring System feedback

C. REQUIREMENTS

The following requirements for YPSim are based on the Mission Description and Task Analysis presented in Chapter III.

1.0 Functional Requirements

1.1 YPSim shall simulate all six YP's training missions.

1.2 YPSim shall allow the user to totally control YP's engines RPM and rudder, allowing free navigation in the mission scenario.

1.3 YPSim shall provide a scenario editing tool for instructor's use.

1.4 YPSim shall provide a Graphical User Interface (GUI) based on video-game style menus and selection of options, allowing the user to intuitively navigates back and forth between initialization, setup and mission execution.

1.5 YPSim GUI shall provide access to menus for (a) mission selection, (b) tutorials for each mission, (c) configuration options, (d) help, and (e) quit.

1.6 YPSim shall provide environmental options at the mission scenario configuration level for (a) daylight, (b) sea state, (c) wind direction and speed, (d) visibility (fog), (e) sea current, and (f) cloud cover.

1.7 YPSim shall present the following viewer options:

1.7.1 First person, from inside the YP's bridge, at three positions (a) amidships, (b) portside wing and (c) starboard wing. The first person view shall be able to control viewing direction in both azimuth and elevation.

1.7.2 Third person, one following the YP with a fixed relative position from astern, and another option with orbital control of the view.

1.7.3 Binocular view when in first person mode (at YP's bridge).

1.8 YPSim shall simulate YP's equipment that represent relevant cues for performing the tasks of Helmsman, Lee Helmsman and OOD. For the Brazilian Naval Academy (BNA) YPs, the following items shall be included, but not limited, to the simulation:

1.8.1 Helm, simulating the wheel movement when rudder orders are executed.

1.8.2 Rudder Angle Indicator, representing the current YP's rudder angle for the helmsman.

1.8.3 Gyro Compass, representing the heading dial movement while turning.

1.8.4 Gyro Compass Repeater, located on each bridge wing (port and starboard).

1.8.5 Engine Control, representing the two levers that control port and starboard engine's RPM. YPSim shall simulate the rotational movement of these control levers providing a visual cue of the current engines RPM status.

1.8.6 Anemometer, simulating port and starboard wind birds with direction and speed of the apparent wind. YPSim shall include turbulence effects, adding noise on direction and speed for the leeward wind bird.

1.8.7 Navigation Radar, simulating antenna, display and basic radar functions. The radar antenna shall rotate at the same RPM and direction of the real YP's radar. The basic radar functions are described as, but not limited to, range, Electronic Bearing Line (EBL) and Variable Range Marker (VRM) controls.

1.8.8 Fathometer, representing the current depth at YP's position.

1.8.9 Speedometer, representing the current YP's speed over the water.

1.8.10 Global Positioning System (GPS), representing YP's current position in latitude and longitude, course over the ground (COG), speed over the ground (SOG), bearing to the next waypoint in route (BRG), and estimated time of arrival to the next waypoint (ETA).

1.9 YPSim shall represent the nautical chart of the current area of operation. YP's current position and orientation shall be represented in this nautical chart by the use of a meaningful icon

1.10 YPSim shall represent an instrument panel, available during the mission execution at every view option. This panel shall consolidate relevant information from, but not limited to, (a) apparent wind, (b) heading, (c) speed, (d) engine RPM status, (e) rudder angle, (f) depth, (g) mission elapsed time.

1.11 YPSim shall simulate YP's wake and bow spray effect, proportional to ship speed, providing visual cue of ship movement to the player.

1.12 YPSim shall simulate Man Overboard (MOB) situations with an animated character, representing the individual, floating in the ocean, waving for rescue. The MOB task shall be activated by scenario triggers or, at any time, by user/instructor command.

1.13 YPSim shall simulate a floating orange smoke marker for the MOB situation. The smoke marker shall last for four minutes from the moment of activation, or be removed from the scene upon the MOB recovery. The activation shall occur by the user's command.

1.14 YPSim shall simulate a standard YP lifebuoy for the MOB situations, respecting its shape, size and colors. The lifebuoy activation shall be given by the user's command, and removed from the scene after recovering the MOB.

1.15 YPSim shall simulate YP's whistle. Whistle activation shall be given by the the user continuously pressing the key or GUI button, and deactivating upon its release.

1.16 YPSim shall simulate flag hoisting in YP's mast by the user's commands. A specific GUI shall provide the basic functionality required for the user's control of flag selection, hoisting and hauling it down.

1.17 YPSim shall provide a scoring system based upon each mission performance metrics. Scores shall be displayed in a debriefing menu, after mission execution. The debriefing menu shall display individual scores per metric evaluated and an overall result by adding all of them in a meaningful way.

1.18 YPSim shall evaluate the user's performance at:

1.18.1 MOB missions by (a) recovery time, (b) MOB safety violation, (c) turning procedures, (d) administrative procedures, (e) recovery distance, (f) recovery side evaluation, (g) number of engine and rudder orders, (h) engine overload, and (i) YP's safety violation.

1.18.2 Anchoring missions by (a) mission time, (b) anchoring maneuvering procedures, (c) administrative procedures, (d) anchoring precision, (e) number of engine and rudder orders, (f) engine overload, and (g) YP's safety violation.

1.18.3 Mooring/unmooring missions by (a) mission time, (b) mooring/unmooring maneuvering procedures, (c) administrative procedures, (d) number of engine and rudder orders, (e) engine overload, (f) YP's safety violation, and (g) number of YP hits at the pier.

1.18.4 Underway Replenishment (UNREP) missions by (a) UNREP maneuvering procedures, (b) administrative procedures, (c) engine overload, and (d) YP's safety violation.

1.18.5 Tactical Maneuvers missions by (a) execution time per evolution in the formation, (b) stationing procedures, (c) administrative procedures, (d) engine overload, and (d) YP's safety violation.

1.18.6 Basic Navigation missions by (a) execution time, (b) rules of the road procedures, (c) offset from the navigation route, (d) engine overload, and (d) YP's safety violation.

1.19 YPSim shall simulate an anchor buoy marking the anchor's position. This buoy's shape, size and color shall match the real buoy used at the YPs.

1.20 YPSim's mission scenarios that simulate real-world geographical areas shall provide accurate terrain geometry and coordinate matching.

1.21 YPSim shall provide mission scenarios with 3D surface contacts representing the real-world types of ships and boats expected for that given area.

1.22 YPSim shall provide mission scenarios with navigational aids, such buoys, lighthouses and lead marks, represented in the simulation. The shape, color and lights of these 3D objects shall accurately match their real-world instances.

1.23 YPSim shall provide an Artificial Intelligence (AI) agent for simulating the behavior of a helmsman steering the YP. A specific GUI for controlling this agent shall be provided, allowing the user to use or not use the AI agent and issue rudder commands.

1.24 YPSim shall provide an AI agent for controlling ships that directly interact with the user's YP. This agent shall control both his ship's course and speed, simulating more realistic movements in the scenario. Ships (usually other YPs) participating in tactical maneuvers and Underway Replenishment (UNREP) operations shall use this agent for movement control.

1.25 YPSim shall handle collision detection between the user's YP and (a) terrain, (b) other surface contacts, and (b) navigational aids.

1.26 YPSim shall provide control of YP's damage level occurred by eventual collisions during the mission execution. A negative reinforcement for every collision shall be applied in the user's score at that given mission. A maximum damage threshold shall be set in the mission scenario configuration, allowing different levels of difficulty for the mission execution.

1.27 YPSim shall provide a realistic ocean surface with sea state (wave height) and color settings in the mission scenario configuration.

1.28 YPSim shall represent navigation routes and waypoints in the 3D scenario, as a reference for the user. This functionality availability shall be set at the mission scenario configuration. When available, this reference shall be switched on/off by user/instructor command during mission execution.

1.29 YPSim shall simulate a wind effect that is initially set in the mission scenario configuration. This environmental factor shall affect YP's physical model, anemometer indicators and wind bird state. A noise component to the wind direction and speed shall be introduced and controlled in the mission scenario configuration.

1.30 YPSim shall simulate a sea current effect that is initially set in the mission scenario configuration. Current's direction and speed shall affect YP's physical model.

1.31 YPSim shall simulate YP's mooring lines for mooring/unmooring missions. A realistic behavior for the mooring lines shall be simulated, affecting the YP's physical model as expected in real life. A specific GUI for mooring

lines interaction shall be provided, allowing the user to give the basic commands for controlling them.

1.32 YPSim shall simulate YP’s anchor and its respective chain, for anchoring missions. Factors such as drifting, weight, chain catenary effect and nature of the seabed shall be included in the anchor’s physical model. A specific GUI for anchor interaction shall be provided, allowing the user to give the basic commands for controlling it.

1.33 YPSim shall simulate YP’s rudder movements, not only representing changes in the rudder angle indicator, but also moving the rudder geometry itself by the current of angle ordered by the user.

1.34 YPSim shall simulate YP’s propellers movements, rotating the propellers geometry at the proper direction and speed, proportional to engines RPM.

1.35 YPSim shall provide network capabilities allowing interoperability of multiple instances of the simulation. The network infrastructure shall be able to connect at least three instances of YPSim running the same mission scenario.

1.36 YPSim shall adopt dead reckoning techniques for more realistic movements of remotely controlled actors in the simulation.

1.37 YPSim shall be initialized in full screen mode without window decoration frame.

1.38 YPSim shall allow the user/instructor to pause the simulation at any time.

1.39 YPSim shall provide After Action Review (AAR) functionality.

2.0 Nonfunctional Requirements

2.1 Usability

2.1.1 YPSim shall provide an entertaining instructional environment related to navigation and shiphandling topics applied to a naval academy's YPs.

2.1.2 YPSim shall replicate YP's geometry and characteristics of maneuver, providing good immersion to its users.

2.1.3 YPSim shall be used with or without an instructor's supervision.

2.1.4 YPSim shall be used as standalone or connected to a network.

2.1.5 YPSim shall provide an intuitive interface, allowing a quick learning process to its users.

2.1.6 YPSim shall have multiple input devices options for controlling the YP platform.

2.1.7 YPSim shall have the option for multiple- or single-screen display.

2.1.8 YPSim shall have an easy installation process.

2.1.9 YPSim shall have a help menu containing instructions about its controls.

2.2 Reliability

2.2.1 YPSim shall be a reliable application, supporting long periods of running time.

2.3 Performance

2.3.1 YPSim shall run in a laptop or desktop PC.

2.3.2 YPSim shall generate, at minimum, 20 frames per second using the recommended hardware configuration listed in section A.5.

D. PRODUCT FEATURES

YPSim's most important features can be summarized into three categories, called System, YP Platform and Scenario. First, we grouped features regarding generic capabilities of the system. The second group represent YPSim characteristics specific for the simulated ship. Features that affect the scenario where the simulation takes place are listed in the last group.

The following list is not complete and represents only the most important features of the system, for synopsis purposes and better understanding of requirement needs.

1.0 System Features

1.1 Simulation of six basic YP's mission

1.2 Intuitive Graphical User Interface (GUI)

1.3 Scoring system

1.4 Network capabilities

1.5 Realistic 3D graphics

1.6 Multiple input devices

1.7 Multiple screen options

1.8 After Action Review

1.9 Multiple view points

1.10 Collision detection

1.11 AI agents for ships in scenario

2.0 YP Platform Features

2.1 Fairly close to reality physical model

2.2 Radar simulation

2.3 Nautical chart simulation

2.4 Anemometer simulation

2.5 Fathometer simulation

2.6 AI Helmsman agent

2.7 Binocular view

2.8 Whistle

2.9 Heads Up Display (HUD) for instruments

2.10 Mooring lines simulation

2.11 Anchor and chain simulation

3.0 Scenario

3.1 Realistic scenarios of the YP's area of operation

3.2 Scenario editing

3.3 Fairly realistic ocean surface

3.4 Environmental control

3.5 Wind effects

3.6 Sea current effects

E. ACCEPTABLE HARDWARE REQUIREMENTS TO USER

The following systems characteristics are required in order to run YPSim with good results:

- **Processor:** dual core CPU @ 2.3GHz, minimum
- **Memory:** 2 GB minimum
- **Disk Space:** 450 MB to install

- **Graphics Card:** 256 MB minimum, with support to OpenGL 2.0 or above (NVidia GeForce 7xxx series or ATI Radeon HD 2xxx series and above)

Operating System: Windows XP or above

V. SYSTEM DEVELOPMENT

A. OVERVIEW

The YPSim is a proof-of-concept application, developed along the two years of the author's master's degree research. Using the same architecture proposed by Salvatore (2005), Ernst (2006), and Toledo (2006) in their designs, YPSim was developed on top of the Delta3D game simulation engine. The underlying details about the Delta3D application and its dependencies is beyond the scope of this thesis and can be found in the previously mentioned theses works. This chapter is dedicated to describe all the steps taken between the idealization and the currently available prototype version of YPSim.

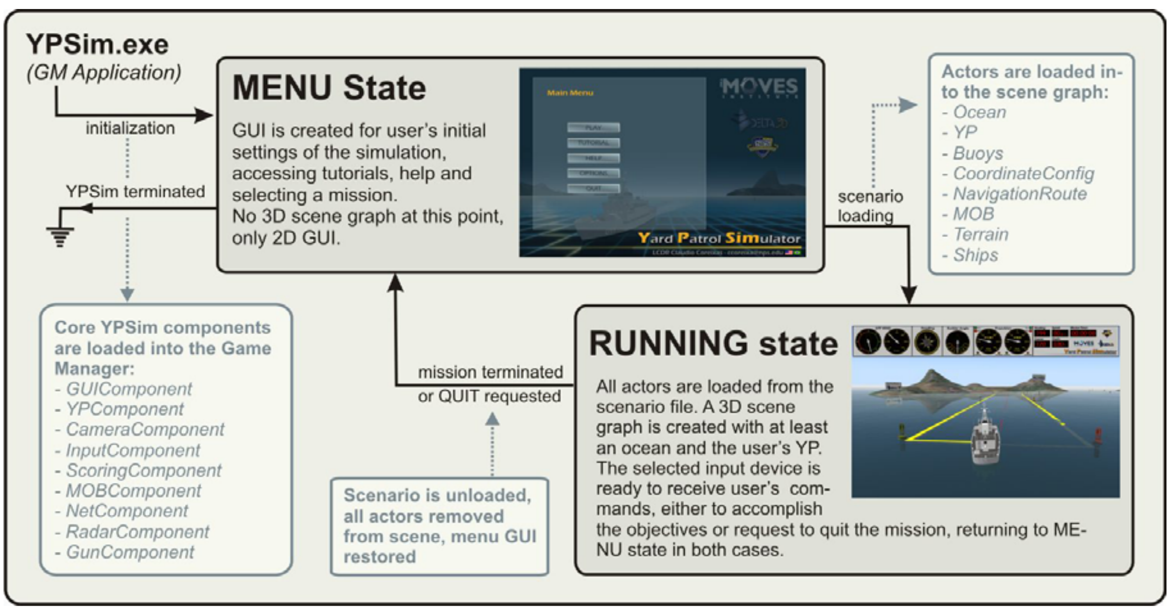


Figure 15. High-level view of YPSim's state cycle.

YPSim is a Delta3D (version 2.4.0) Game Manager (GM) application (Toledo, 2006) that loads some important GM components and one selected scenario file containing, at a minimum, one Ocean Actor and one Yard Patrol actor (YPActor). At the initialization of YPSim, components with specific code

functionalities are added to the GM, such as the one responsible for the creation of a graphical user interface (GUIComponent on Figure 15.). After initialized, the YPSim will present a GUI to the user, where he/she can navigate throughout the many menu options available, selecting options, accessing tutorials, help and mission selection. Having concluded all the menu selections desired, the user selects one combination of (a) YP, (b) environment, and (c) mission to play, and loads the scenario file. The scenario file sets the stage for the simulation with the actors, such as user YP, ships, terrain, ocean, and environment, describing their specific properties for that mission. Once the scenario is loaded, the user is ready to control his/her YP in order to accomplish the goal for that mission. Figure 15 presents a high-level view of the YPSim's states cycle, giving a graphical interpretation to the sequence described in this paragraph.

The biggest constraint to the YPSim's prototype development cycle was time. The author's lack of programming background, 3D graphics creation, and class obligations limited the number of features implemented in this game-based simulation project. YPSim beta version 0.13, the proof-of-concept prototype, has met most of the requirements listed in Chapter IV. For the scope of this thesis, only the Man Overboard (MOB) and Anchoring missions were implemented in YPSim v0.13, using two scenario environments. The first environment represents the surroundings of the Brazilian Naval Academy (BNA) at the Guanabara Bay (Rio de Janeiro – Brazil), while the second is a set of fictitious islands at open sea. Some important functionality, such as After Action Review (AAR), could not be achieved due to development time limitations and had to be listed as future research work. The following sections describe the development of the most important pieces of YPSim v0.13.

B. 3D MODELS

YPSim uses OSG and IVE file formats for the 3D meshes loaded into the simulation. These file formats are the most commonly used formats in applications developed using Delta3D and OpenSceneGraph (OSG). While OSG

are text files containing the model information and are relatively easy to manipulate in any text editor, IVE files contain the same information and textures images in a binary format, making the loading process more efficient. The 3D models used in YPSim come from three different sources:

- Created by author
- Delta3D asset library
- *Laboratorio de Sistemas Digitais (LSI)* – USP (Digital Systems Laboratory, University of São Paulo)

We used X3D (Extensible 3D Graphics) Editor, Blender and 3D Studio Max™ software for model creation. Models from the Delta3D asset library were ready to use, in both OSG and IVE formats, and often with the source 3D Studio Max file available for further editing. The LSI's models were initially created using Autodesk Maya™ for another simulation project for the Brazilian Navy. These models were not OSG ready and required further manipulation using 3D Studio Max™. Appendix A contains detailed information about the 3D models used in YPSim.

C. YP'S PHYSICAL MODEL

The physical model is one of the most important pieces of YPSim's code. Its origins were in an application developed in Delta3D for demonstrations purposes called WaterSprint. The WaterSprint was implemented by Erik Johnson, the Delta3D's software engineer, in order to create a demo application to explore the dtOcean library used to simulate ocean effects. WaterSprint had a physical model implemented to simulate a rigid inflatable boat (RIB) with an outboard motor moving over the ocean according to the user's input command. Its physical model was fairly simple and efficient in controlling the RIB with realistic movements. The initial idea was to reuse, as much as possible, pieces of code from the WaterSprint model into YPSim, refining the model one piece at a time, according to the YP's characteristics.

Our physical model, detailed in Appendix D, uses Open Dynamics Engine (ODE) and its correspondent Delta3D rigid body wrapper functions to apply forces on the YP at each physical simulation step (1/1000 seconds step size). This model is not designed to be complete, and several simplifications were made in order to achieve real-time performance, with realistic behavior for a game designed to offer a very basic type of training.

D. YP'S MOORING LINES

Mooring lines have an important role during mooring and unmooring maneuvers, being responsible for pivoting and securing the ship to the pier. Learning how to handle a ship using mooring lines is one of the most important types of training aboard the YP. The scope of this thesis is directed towards the Man Overboard and Anchoring missions, which do not require the use of mooring lines. However, because of its importance for future implementations, basic mooring lines functionality was added to the YPSim. Detailed information about the mooring lines implementation are given in Appendix B.

E. YP'S ANCHOR AND CHAIN

An anchor model was developed for YPSim's Anchoring missions, allowing the simulation of the physics involved in that task. Anchoring the YP is not a simple process and its complexity is a product of many important variables, such as wind, current, nature of the seabed, and OOD's ship-handling skills. A model capable of simulating some of the basic dynamics of the anchoring process is crucial in providing valid training for YPSim's Anchoring mission. More detailed descriptions of the anchor and chain model are provided in Appendix C.

F. OCEAN MODEL

YPSim uses a dtOcean actor in each one of its scenario files, for environment control and realistic ocean rendering. The dtOcean is a Delta3D actor that contains objects of the osgOcean (version 1.0.1) library. The scenario

configuration files can set several parameters of the ocean surface, such as resolution, choppiness, wind speed, height, reflection, water silt and color. The current version of dtOcean used in YPSim, also allows the configuration of sea current parameters that are used in the YP physical model calculations.

G. GAME COMPONENTS

YPSim is a game application, subclass of the dtGame::GameApplication class. Game applications are characterized by the use of the Delta3D’s Game Manager (GM) message system to exchange important information between specific components, as explored by Toledo (2005) in his design. GM components used in YPSim are subclasses of dtGame::GMComponent, designed to provide well defined functionality to the simulation, using the GM infrastructure. One of the biggest advantages of the GM infrastructure when adding functionality to the code, is the facility in which new pieces can be added without changing other components. The message system, used as a means of communication between components, provides an excellent tool for connecting the major pieces of the simulation.

YPSim has eleven major Game Components in its architecture:

- **YPComponent:** responsible for directly handling messages and settings to user’s YP, an instance of the YPActor class.
- **CameraComponent:** responsible for handling camera position and modes of operation, handling messages concerning the scene camera. The camera corresponds to user’s eye on the scene.
- **GUIComponent:** responsible for the Graphical User Interface (GUI). Handles messages that affect the GUI and sends messages based on user inputs on the GUI options.
- **HelmsmanComponent:** corresponds to the helmsman agent. Has a great interaction with the YPComponent, sending commands to control YP’s rudder. This component has its own GUI to handle user’s inputs and information output from the helmsman Artificial Intelligence (AI).
- **InitComponent:** responsible for basic initialization settings on the application, when the mission scenario is changed.

- **InputComponent:** responsible for managing the input devices. Handles messages to set the selected device, configuring it, and processing user's commands. Always handles inputs from keyboard and mouse, even when other devices are selected as the main controller.
- **NetComponent:** responsible for handling network connections, and packets reception and transmission. Creates and update the YPs corresponding to the other instances of YPSim connected on the network. YPSim uses a Client/Server connection, using specific UDP protocol to send its YP current state.
- **MOBComponent:** responsible for managing the Man Overboard (MOB) task and the character animation (a simple hand waving for rescue). This component handles the conditions to drop and recover the MOB, sending and receiving messages necessary to execute the task.
- **RadarComponent:** manages the radar simulation. Basically handles messages for controlling the radar, such as on/off, range, Electronic Bearing Line (EBL) and Variable Range Marker (VRM). Also controls the radar screen appearance, if inside the bridge, as a regular radar, or Heads-up Display (HUD).
- **ScoringComponent:** responsible for assessing user's performance in a given mission. Calculates user score according to the type of mission and scenario executed. Handles messages to terminate mission and create debriefing menus.

H. YPSIM ACTORS

1. YPActor

The YPActor is the main actor present on YPSim. Actors are the entities that interact in the simulation scenario, and the YPActor is the only one under the user's direct control. Every scenario configuration file must contain one instance of the YPActor class, representing the user's YP. This class is a subclass of the dtActors::GameMeshActor that contains a set of specific objects and methods designed to simulate a fully functional YP controlled by the user.

A high-level view of the YPActor class can be visualized in Figure 16, presenting the most used public methods available for the Game Components, in this case YPComponent and InputComponent. The YPActor owns instances of

the Buoy, YPPhysical, MooringLines, Anchor and YPEffects classes. Each one of this objects will be responsible for a specific functionality inside the YPActor, accessing important information and other objects inherited from the dtActors::GameMeshActor class. All of them need to access the ODE body inside the GameMeshActor in order to apply forces, retrieve velocity information or create joints. Others, like YPPhysical and MooringLines, need to access nodes inside the 3D Model to apply changes in the geometry to be rendered on scene.

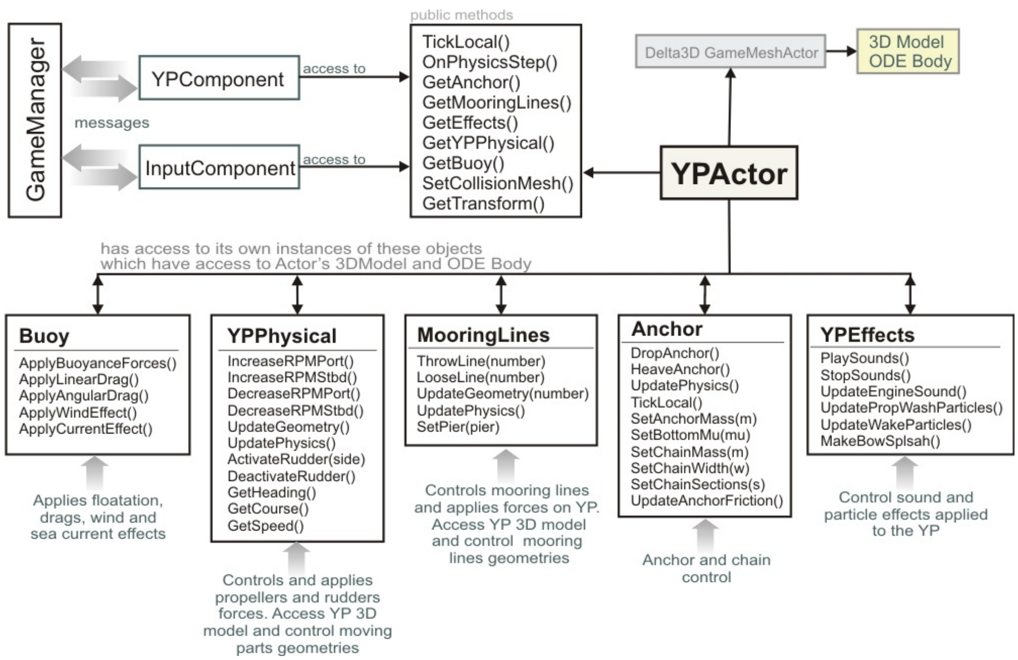


Figure 16. High-level view of the YPActor class.

2. ShipDummyActor

The ShipDummyActor class is used for creating nonrealistic ships on the scenario. A course and speed can be configurable on the mission scenario file, and the this actor will move in a straight line indefinitely during the runtime. This ship actor is not intended to handle collisions, ocean response and course, or speed control, making it computationally simple to populate the scenario.

ShipDummyActor is a subclass of dtActors::GameMeshActor and allows the user to select which 3D model to represent it on the scene.

3. ShipSmartActor

When more realistic behavior is expected for a ship, the ShipSmartActor can be used. Actors of this class have a small physical model that will handle ocean effects (buoyance), and course and speed control, and have sound and particle effects to simulate the ship's engines, bow splash and wake. This yields to a good and realistic behavior that can be used to simulate ships that will directly interact with the user's YP, usually visualized at closer distance. The ShipSmartActor class was primarily created to simulate other YPs on tactical maneuvers or Underway Replenishment (UNREP) missions. In both situations, ships can change course and speed during the runtime, and will be at close distance to the user's YP. The drawback of using ShipSmartActors on a scenario is the intensive processing involved in its physics step, more than a simple ShipDummyActor and less than YPActor.

I. GRAPHICAL USER INTERFACE (GUI)

Developed using the dtGUI and CEGUI libraries, a graphical interface was created in order to allow user interaction with the simulation. YPSim has basically two states during its execution: (a) menu, and (b) running.

Most of the GUI functionality is used at the menu state, when the user is navigating through the options available on the screen, using a mouse. Using the menus available, the user can select configuration options for screen type, input device and mission selection. He/she can also access tutorials and help information menus, the contents of which were not implemented for the scope of this thesis.

Figure 17. Screenshot of YPSim's GUI menus.

During the running state of the simulation, GUI functionality is available for controlling the Helmsman AI, quitting the current mission and debriefing information. During the mission execution, information about the YP's instruments are displayed using a different technique. Instead of using dtGUI/CEGUI for displaying instrument information, the author used 3D geometries with DOFTransform and Switch nodes. These OSG special nodes allow manipulation on objects rendered geometries, giving extra flexibility for creating effects such as radar screen, nautical chart, and dials and digits on the instrument panel (Figure 18).

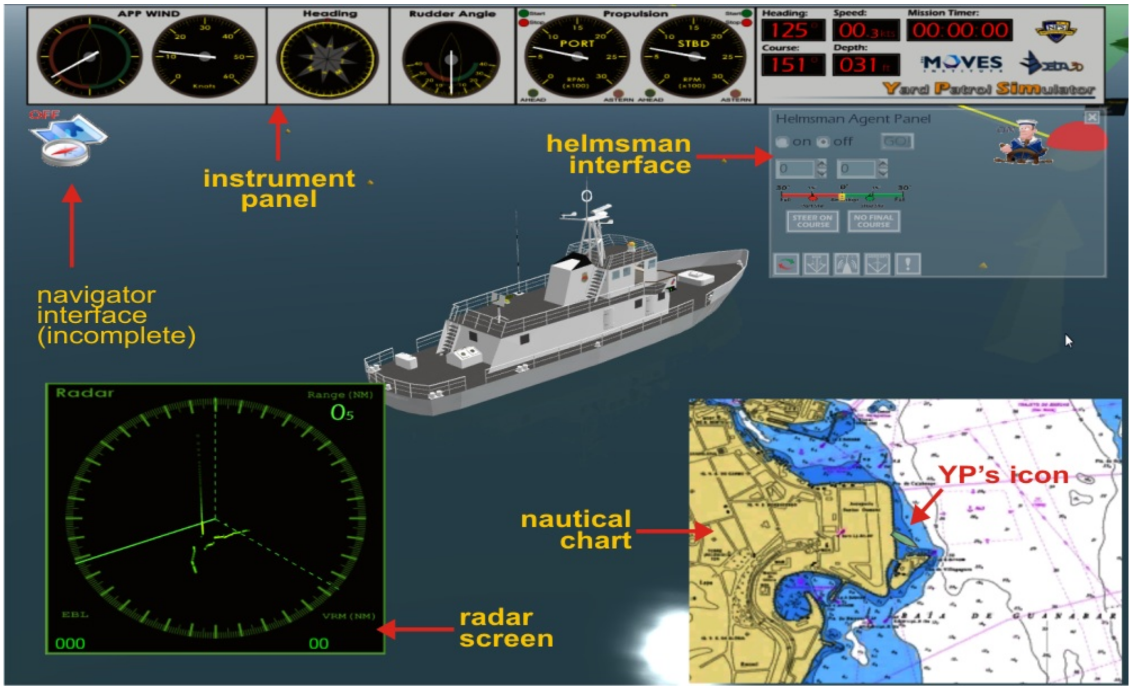


Figure 18. Screenshot of YPSim's GUI at runtime.

J. NAVIGATION RADAR

We modeled a simple navigation radar found inside the YP's bridge in order to simulate the basic functionality of this important piece of equipment. The radar model has the capability of measuring bearings and distances of landmarks and surface contacts. The Officer of the Deck (OOD) uses these radar functions for most of the missions executed in YPSim. Appendix E presents more detailed explanations of the radar implementation.

K. NAUTICAL CHART

In order to simulate the positioning information provided by the Quartermaster of the watch (QOW) or GPS plotter, a nautical chart functionality was developed. The virtual nautical chart can be displayed on the screen by the user's request, showing a YP's icon at the center of the chart oriented to its present head. The chart itself is an instance of the Detla3D dtCore::Object with a

special 3D model loaded as a geometry. This model contains a flat plane with a texture of the simulated nautical chart with the YP's icon. The icon plane has a DOFTransform node on top of it, allowing runtime manipulation for heading control. By manipulating the texture coordinates of the nautical chart geometry during the runtime, we were able to update the YP's icon position and provide zoom control.

L. INPUT DEVICES

The standard input devices used by YPSim are keyboard and mouse. In order to explore YPSim as a game and provide more flexibility to its usability, other common types of devices were configured to provide commands to the user's YP.

Figure 19. YPSim input devices. From left to right: keyboard/mouse, flight yoke, Ship Driver Controller(TM), and PS3 controller.

M. CAMERA VIEWS

Four basic camera view modes were implemented, allowing different perspectives of the scene during the mission execution. This flexibility is important in order to allow users to visualize the environment from different angles. The camera modes are as follow:

- **First person:** user’s eyes are located inside the YP’s navigation bridge. User can select three different locations inside the bridge, left wing, centered, and right wing. A binoculars view is available during this camera mode, magnifying user’s field of view.
- **Follow:** users have a third-person view of the YP, maintaining fixed relative position at the YP’s stern.
- **Orbit relative:** users also have a third-person view of the YP but controlling the relative position (bearing and distance) to the YP. This mode allows the user to keep looking the YP at the same angle during turns.
- **Orbit true:** has a similar effect to relative, but keeps the same true bearing and distance to the YP. In this mode, if the YP turns, the user will see it from another angle, making it easy to notice changes in course.

N. SCENARIO CREATION

Scenario configuration files, also called maps, contain the actors’ properties composing the simulated mission. These files are edited in a Delta3D application called STAGE and saved in a XML format, under the .map extension. STAGE makes it easy to change an actor’s basic parameters, such as position and orientation, or more complex settings particular to a given class, such as the sea current direction of the ocean actor (Figure 20). Maps are loaded after the user selects the START button under the mission selection menu, changing YPSim to the running state, and unloaded every time the simulation goes back to the menu state.

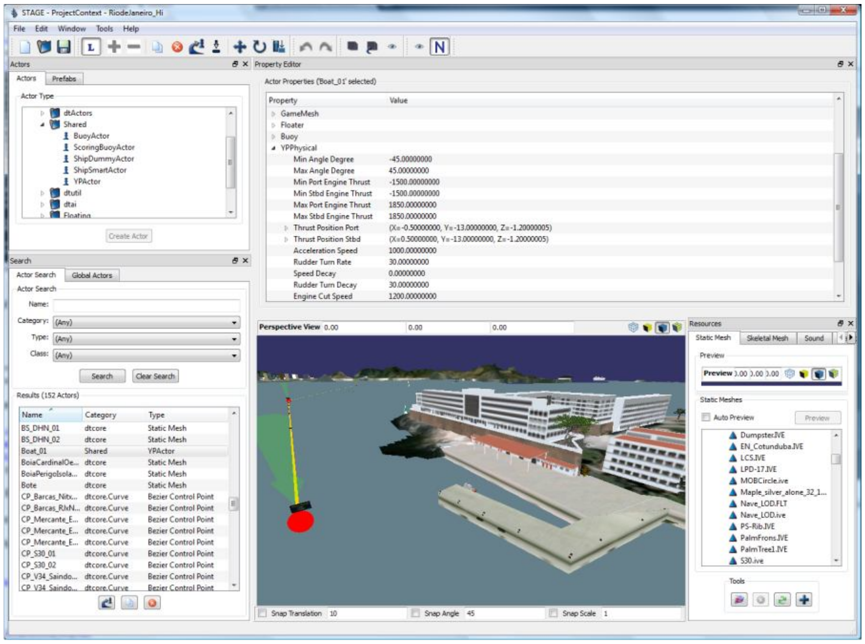


Figure 20. Screenshot of a scenario editing using Delta3D's STAGE application.

Ten scenario configuration files were built for YPSim, each one representing a combination of two environments (Guanabara Bay or Islands at Open Sea) and five missions (Anchoring, Navigation, Replenishment At Sea, Man Overboard and Mooring). Each map has a specific configuration for wind, current, ocean color, surface contacts present on the scene, and tasks to accomplish. For the purpose of this thesis, only the Man Overboard (MOB) maps were completely implemented with the tasks and scoring elements. Maps for the other missions exist, but do not have the tasks part.

O. SCORING SYSTEM

Scoring systems for YPSim should follow the instructions for performance assessment presented in Chapter III of this thesis. The system should evaluate user actions according to metrics relevant to each specific mission. For the scope of this thesis research, only the MOB mission had the scoring system implemented.

The MOB scenario consists of a simple navigation route marked by several pair of buoys and a final waypoint where the MOB will be dropped upon arrival. The user will first navigate along the route, trying to stay as close as possible to the intended route and reach the MOB waypoint as fast as he/she can. Once the MOB is dropped, the user has 150 seconds to recover the individual. Mission ends under two conditions: when MOB is picked, or after 150 seconds of rescue time. The scoring system for the MOB mission is measuring the elapsed time of the navigation and rescue part, the number of input commands (rudder and engines) and the distance offset from the center of the route. Other performance metrics established for the MOB mission on the YPSim's requirements were excluded from the scope of this thesis.

VI. YPSIM TESTING

The development of YPSim was a continuous process that took two years to achieve the current proof-of-concept version (YPSim v0.13). Considering the importance of keeping a user-centered design for this game-based simulation, we decided to make several releases along the way, enabling feedback from users. The feedback was used to correct design issues, code verification and improve usability of the application. We used four milestone releases of YPSim to achieve our goal of having a polished proof-of-concept prototype.

The first release occurred during the 10th Annual MOVES Research and Education Summit open house event (July 14, 2010), when a demo of the first release of YPSim was made to the public (YPSim v0.1). The second milestone was a demo of YPSim v0.5 at the Interservice/Industry Training, Simulation and Education Conference (I/ITSEC) 2010, five months after the previous release (December 2010). The next release was made during the following MOVES open house in 2011, when version 0.11 was demoed in July 13, 2011. Finally, after having polished the last prototype version of the simulator, YPSim v0.13 was released at the Brazilian Naval Academy (BNA) for the user acceptance survey described in Chapter VII.

A. MOVES OPEN HOUSE DEMO RELEASE (2010)

In July 2010, the first stable prototype of YPSim was released for public use during the MOVES open house event of the Annual Research and Education Summit. This version was the first chance to test the basic simulation platform, consisting of the ocean model, a coarse physical model implementation for the YP, and the terrain 3D model of the Guanabara Bay (Rio de Janeiro, Brazil), including BNA’s surroundings. This public release represented our first chance to get feedback from the modeling and simulation community regarding the YPSim concept of operation.

The demo was configured using YPSim running in a desktop computer connected to the MOVES Institute three-screen CAVE© projection system and a flight yoke controller. The selected visualization interface, using a CAVE, allowed users to have a 180-degree horizontal field. Volunteer users were allowed to handle the virtual YP through a green line on the scenario, representing a navigation route to the Brazilian Naval Academy, as shown on Figure 21. The nautical chart was made available for this demo, representing both the YP's current position, and heading and waypoints along the route. Using the flight yoke controller, the players were able to control the ship's course and speed over the simulation, while checking the YP's indicators on screen.

Figure 21. YPSim demo at the 2010 MOVES open house event.

The general impression of this first release of YPSim was very positive, and the feedback collected was of great value to improve the software. The physical model was not yet refined, and the ship's response to the commands were not realistic, generating important observations from most of the naval officers who drove the virtual YP. We also observed that some lag effects were created while loading the BNA 3D model, whenever it became visible on the screen. Posterior investigation on the BNA 3D model led to an optimization on its geometry, significantly reducing its file size and loading time. Since the demoed

version of YPSim was not overly sophisticated, only basic functionality and training concepts were tested, but with great results.

B. I/ITSEC DEMO RELEASE

In December 2010, five months after the MOVES open house release, we demoed YPSim v0.5 at the exposition floor of the I/ITSEC 2010. This version of YPSim presented significant improvements compared to the previous release, including a Graphical User Interface (GUI), a helmsman artificial intelligence (AI), radar simulation, and refinements on the physical model. An important bug found on the ocean model was also removed, making the simulation more stable and robust. YPSim was made available for general use during four days of conference running on the author’s personal laptop computer and displayed on a 50-inch LCD TV. A Playstation3™ controller was used as the main input device for controlling the YP.

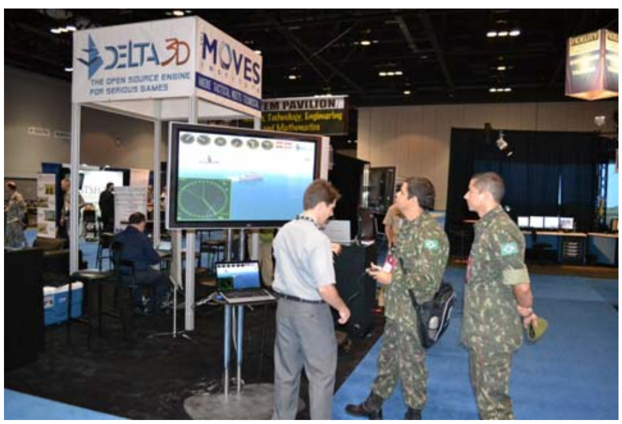


Figure 22. YPSim demo at the I/ITSEC 2010.

The demo attracted approximately one hundred users during the four days of public exposition at the I/ITSEC, almost all of them with a moderate to high level of expertise on the training simulation field. The public response was, again, surprisingly positive, with good questions being asked about YPSim implementation concept and the Delta3D game and simulation engine. One of the most interesting questions was one regarding the similarities with the

commercial software Ship Simulator™, developed by the Dutch company VSTEP. Ship Simulator, the commercial game described in Chapter II, is a very generic application that provides a virtual environment of a civilian boat (i.e., tug boats, transatlantic ships, cargo ships, water taxi, powerboats, jet skis) and a mission that is not related to the instruction of shiphandling skills learned at naval academies and, therefore, does not fit our needs. In order to do that, we need a game-based simulation that has a very specific context, simulating the YP, doing the same type of missions we do aboard these boats (YPs). A brief explanation of the task analysis conducted in Chapter III helped users to understand the conceptual difference, and the importance of having a game-based simulator specifically designed for the midshipmen world, specially in the military scenario.

C. MOVES OPEN HOUSE DEMO RELEASE (2011)

Version 0.11 of YPSim was released during the 2011 MOVES open house, in July 2011. This version contained major changes in the physical model of the YP, significantly improving the realism of the ship's movements and the user's commands. Mooring lines and the anchor models added extra functionality to the simulator, allowing users to moor/unmoor the YP on a pier and also perform anchoring tasks. The feature most explored during this demo was the scoring system and the Man Overboard (MOB) mission configuration, where users could track their performance on a score ranking system updated every trial. This feature helped to attract users to try YPSim, inducing an informal competition between them in order to put a name at the top of the rank.

For this demo, we used two instances of YPSim running at the same time in different desktops. Each station was using a regular 19-inch LCD monitor with a Ship Driver™ controller as input device. This two instances were loading YPSim from a local server, sharing the same score data file, which was used to keep track of the top ten best scores on the MOB mission to be played. Each player started the MOB mission at the beginning of a navigation route and, as quickly as possible, reached a final waypoint that triggered the MOB event. The

player's score was calculated using the time to accomplish the mission and offset from the navigation route as performance measures. Upon the MOB recover, the score was calculated and compared with the top ten rank list, displaying the results in a debriefing menu. If the score was good enough, it was placed on its rank position and the system asked for player's name.

The improvements on the YPSim interface, stability and physical model refinements generated a significantly less number of critics and better reviews from the public, giving us confidence going into the next step on the release process. Only a few minor adjustments appeared critical to perform before making the final test with the BNA midshipmen.

D. FINAL BNA RELEASE

The final release of the prototype version of YPSim (v0.13) was made on August 1, 2011. This was the most critical step on the release process, now involving the software's final user population, the BNA midshipmen. This would also represent the first impression of YPSim to the academy's instructors and officers, who are familiar with the real BNA YP, the simulated platform. Feedback from this release was collected in a formal survey conducted with the midshipmen during a two-week trial of YPSim running in a lab. The lab setup included a three-screen (120-degree horizontal FOV) display, flight yoke controller and the use as a team (1 officer of the deck, 1 helmsman, 1 lee-helmsman), as shown in Figure 23. BNA instructors asked for a configuration closer to the real YP situation, rather than a game approach, which led to YPSim running on a first-person view mode, inside the bridge, almost all the time.

YPSim version 0.13 was tested during two weeks by 40 midshipmen of the 2nd, 3rd and 4th grades. These midshipmen were volunteers to test the simulator and all had, at least, initial basic classroom instruction of navigation and shiphandling. Most of the users for this release had also good experience on navigation and shiphandling the training at sea, using BNA YPs. Users were not required to have any proficiency level on using computer and/or games.

Experienced BNA instructors and YP Commanding Officers were also invited to freely use YPSim in this release, providing valuable feedback to improve our design. The BNA Superintendent and the Dean of Education actively participated on this release, demonstrating great institutional support to our research efforts.

The results of BNA and midshipmen and instructors exposure to YPSim were incredibly positive. Both sides confirmed the potential use of YPSim in the current instructional framework for navigation and shiphandling topics at the BNA. As a feedback, we were able to collect valuable users' critics about the graphics quality, interface and physics model that pointed some weakness of the current prototype. After the release of version 0.13 at the BNA, the academy command requested to keep using the simulator as an experimental training resource for pre-sail familiarization before the YP classes.

Figure 23. BNA midshipmen driving the virtual YP during the final release of YPSim.

VII. USER ACCEPTANCE SURVEY

We conducted a formal user acceptance survey in order to assess important product feedback from the intended user population of YPSim. After having approved the IRB process to conduct a survey at the Brazilian Naval Academy (BNA), we released the final prototype version of YPSim (version 0.13) to a sample of forty volunteer midshipmen. A navigation and shiphandling instructor at the BNA, who is also a former Commanding Officer of a BNA YP, integrated our research team and conducted this survey in Brazil. We designed a specific survey questionnaire (Appendix F) to collect demographics information and midshipmen's opinions of the current BNA learning framework and YPSim usability. This chapter describes the method used for this study and includes a brief analysis of the results found.

A. METHOD

The methodology used in this study was based on providing controlled exposure of YPSim to a sample of its end user population at the BNA and collecting their reviews in a user-acceptance questionnaire, after the using the system.

1. Participants

The study collected data from a sample of forty volunteer BNA midshipmen, ranging in age from 18 to 23 years. All participants were male, since the BNA does not have any female midshipmen. We asked for volunteers among the three senior grades on the curriculum, corresponding to the 2nd, 3rd and 4th year midshipmen population.

2. Materials

We installed YPSim v0.13 on a desktop with the following hardware specifications:

- CPU: Intel Xeon 2.53GHz
- Graphics card: NVidia Quadro FX 580 (512MB)
- RAM: 12GB
- Operating system: Windows 7™ 64-bit

A Saitek® Pro Flight Yoke System was used as main input device for YPSim, while three 50-inch LCD TVs were used to display the simulation, offering 120 degrees of horizontal Field of View (h-FOV) to the users. We used a Portuguese translation of the questionnaire described in Appendix F to conduct the user acceptance survey. The system demo was set up at the BNA shiphandling and navigation lab.

3. Task

Users were requested to use YPSim as a small bridge team, playing the roles of officer of the deck (OOD), helmsman and lee-helmsman in a basic navigation mission. A fourth midshipmen was asked to just observe the simulation trial, not interfering in the maneuver. The OOD midshipman was in charge of handling the ship along a given navigation route, at the BNA surroundings, providing rudder and engine orders respectively to the helmsman and lee-helmsman midshipmen. At the end of the navigation route, the group was allowed to freely perform MOB and mooring tasks, exploring the system functionalities.

4. Procedure

The forty volunteer midshipmen were randomly divided into ten groups of four and requested to, as a group, use YPSim for a single 50-minute session. Trial groups were randomly scheduled to perform the YPSim trial in a 70-minute block at the afternoon, after regular classes, during two weeks (one group per day). Our investigator at the BNA conducted an initial 10-minute briefing about the YPSim concept, use and the task to perform. Participants were allowed to ask questions at any time during the briefing and trial session.

After the briefing, midshipmen of each group were assigned to play the four roles previously described, respecting the same distribution found aboard the YP. Senior midshipmen were assigned to OOD and observer roles, while the junior midshipmen assumed the helm and engine controls. After 25 minutes of trial, OOD and observer switch roles, allowing both of them to actively experiment YPSim. During the trials, YPSim was used in the first-person view mode, from inside the bridge. After using the simulator, participants were asked to fill out a post-trial questionnaire and were then dismissed.

B. RESULTS

The questionnaire answers are summarized in Appendix G. We made an analysis of the distribution for each answer obtained, hoping to find evidence to support some assumptions made about the YP system and YPSim's usability. Answers for the Likert scale correspondence range from 1 to disagree with the proposed statement and 5 to a full agreement. To provide a general picture of the results observed in each Likert scale question, we provided a bar graph representing their average answer. However, the complete understanding of end user trend should be inferred from the distribution plot, since the average can mask important patterns in the response, such as mode and distribution.

1. Demographics

Among the 40 participants, 15 were from the 2nd year, thirteen from 3rd year, and twelve from 4th year grades on the BNA midshipmen rank. Only 12.5% of them had collateral duties as navigation and shiphandling mentors aboard the YPs. The proportion of reported gamers found was 37.5%—less than expected given other surveys made in the U.S. for the same age range in the military community. Only 15% of the participants reported having previously played any ship simulator type of game, indicating that the user population is not extremely familiar with this type of technology.

2. YP Learning Framework

Supporting our initial assumptions that midshipmen require a significant amount of familiarization time during the hands-on training aboard the YP, 97.5% of the participants answered that they required between 10 and 60 minutes to get used to the task. Although this is a high percentage of the sample population, the mode of this distribution was between 10 and 30 minutes (67.5%), considered lower than the expected mode (between 30 and 60 minutes). The preferred learning method used by the midshipmen aboard was largely divided between *observe* and *comfortable to ask questions* and *doing by himself* (40%) and *not afraid to fail* (42.5%). This distribution shows that midshipmen, in general, feel free to interact with the system, asking questions and not afraid to make mistakes.

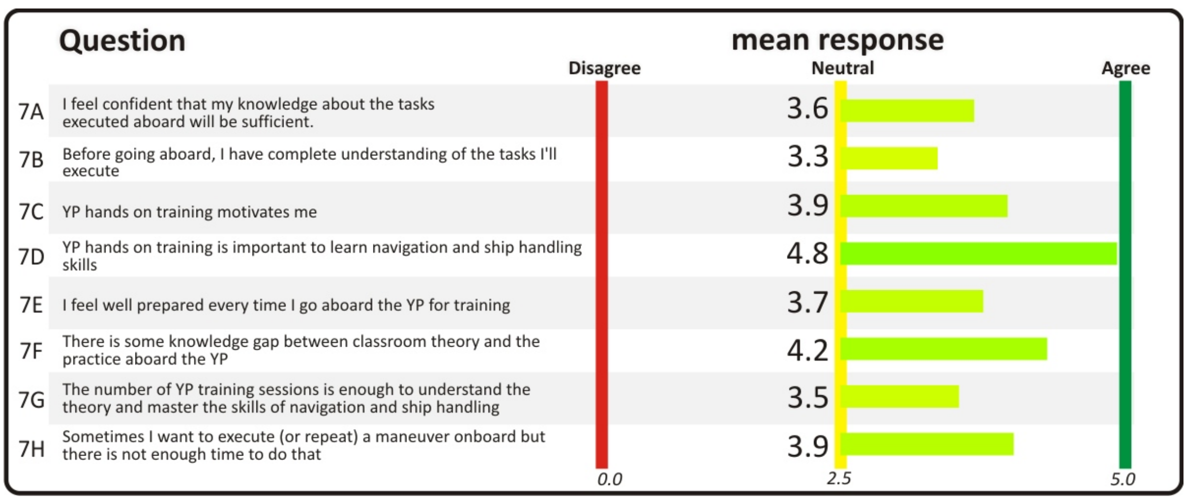


Figure 24. Mean response diagram for questions about the current learning framework.

Questions about the midshipmen's opinion of the YP framework revealed that their confidence over having enough previous theoretical knowledge is not very high. With a similar distribution, midshipmen reported not having total previous understanding of the tasks to be conducted in a hands-on training aboard the YP (question 7B, mean 3.3 and mode 4 with 40%). When considering

the YP training as a motivational activity, most of the participants agreed, in some degree, with the statement (question 7C, mean 3.9 and mode 5 with 35%). The numbers proved that midshipmen believe YP training is an important step in learning navigation and shiphandling (question 7D, mean 4.8 and mode 5 with 80%). In general, the midshipmen do not feel confidently prepared for the YP training events (question 7E, mean 3.7 and mode 4 with 40%) and 57% of the participants fully agreed with the presence of a knowledge gap between classroom theory and practice aboard. Evidence that the current number of hands-on training sessions is not enough is suggested given, the distribution found for this question (question 7G, mean 3.5 and mode 4 with 45%). The answer distribution also supports the assumption that midshipmen, in general, would like to have more maneuvering opportunities, restricted by the time (question 7H, 42.5% of fully agreement).

3. Use of a Simulator for Training

Most of the participants agreed to some degree that the use of a ship-driving simulator would help to reduce any knowledge gap between classroom and hands-on training (question 8A, mean 4.1 and modes 4 and 5 with both 37.5%). About the use of a ship-driving simulator being motivational to the body of students, 80% fully agreed with this statement. This is extremely relevant to support the assumption that the user population is prone to accept this type of technology. The use of a ship-driving game simulator before the aboard tasks on the YP was fully supported by 75% of the participants. For the statement that a YP simulator is not important and will never help to improve performance, 90% of the participants fully disagreed with this idea. Most of the midshipmen (82.5%) considered the use of a YP simulator as a valuable instructional resource inside a classroom, providing a better understanding of the theory.

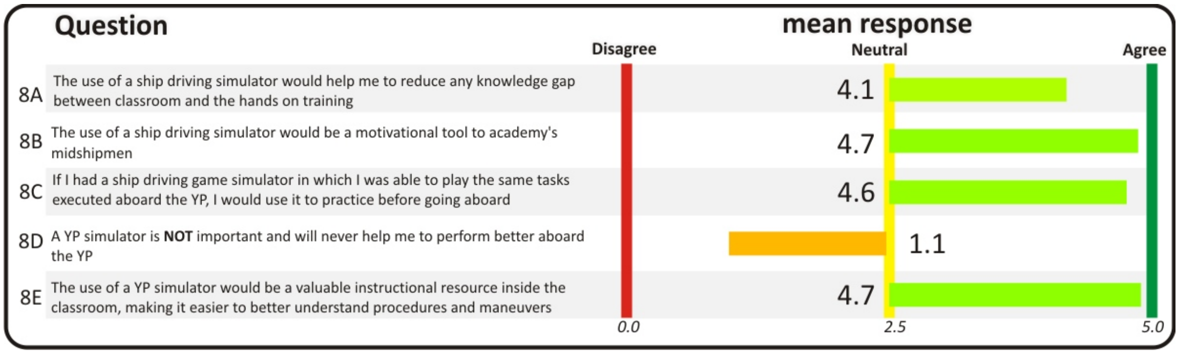


Figure 25. Mean response diagram of questions about the use of simulators for training.

4. YPSim Usability

The initial impression of the participants about the graphic quality of YPSim was good (question 8F, mean 4.0 and mode 5 with 47.5%) even though the trialed version was only a prototype without refinements on the 3D models used. When asked about YPSim being a valuable training tool, most of the participants (62.5%) agreed with the statement. The present sensation of YPSim did not score well among the midshipmen, and the distribution for this statement was evenly spread without a significant mode on the level of agreement. When asked about playing YPSim on their own PCs, if available, participants agreed with the statement (question 8I, mean 4.0 and mode 5 with 40%). Although this was a good indicator, the distribution was less positive than expected when compared with answers found for a previous similar question (8C) regarding the personal use of a ship-driving simulator.

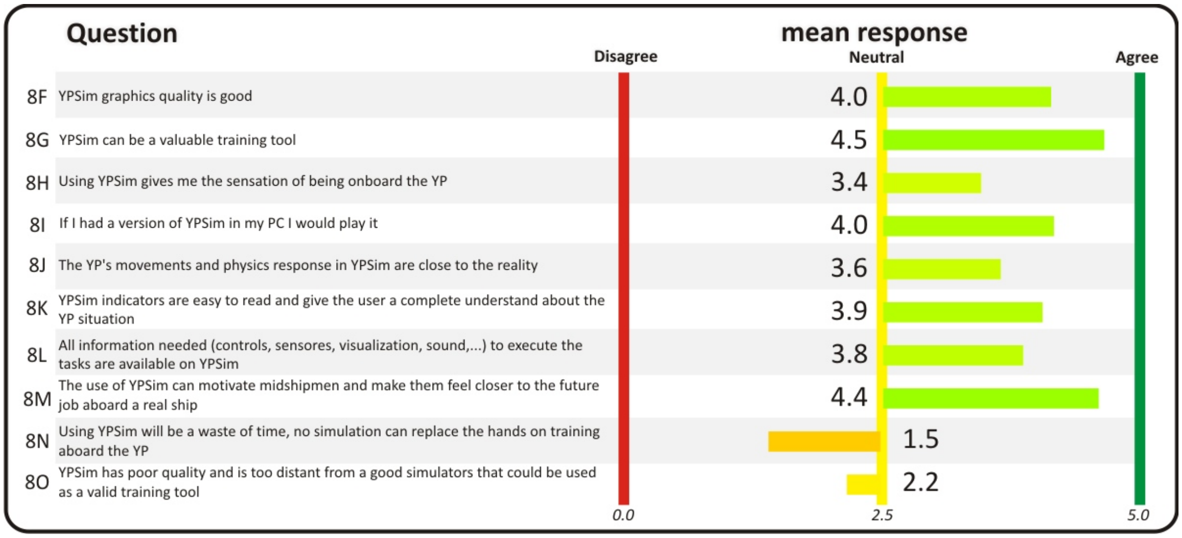


Figure 26. Mean response diagram of questions specifically about YPSim usability.

The level of agreement with the realism of the YP's physical model in YPSim was moderately good (question 8J, mean 3.6 and mode 4 with 45%), with similar distributions to the interface quality questions 8K and 8L (means 3.9 and 3.8, mode 4 with 47.5% and 60%, respectively). Most of the participants fully agreed with potential of YPSim as a motivational asset to their careers (question 8M, mean 4.4 and mode 5 with 67.5%). With almost the same reverse distribution, the use of YPSim was not considered a complete waste of time by 67.5% of the participants. When compared to other simulators that participants could have in mind, YPSim was evaluated as not too distant to other technologies (question 8O, mean 2.2 and mode 2 with 37.5%).

5. Open-Ended Questions

From the sample of forty participants, only 23 answered the open-ended questions requesting comments and suggestions to improve YPSim. The answers provided valuable feedback about the user's opinion of the simulator, providing good insights into future steps for implementing it as a training tool at the BNA learning framework. About six participants suggested the importance of

having YPSim freely distributed among the midshipmen body of students for use on their personal PCs. Other participants' responses reinforced the motivational aspect of using YPSim as a training tool, more specifically to the freshmen midshipman of the 1st and 2nd years, complementing the initial basic professional courses of the BNA. In general, participants who responded the open-ended questions demonstrated enthusiasm with the application and were excited about using it in the near future. One midshipman suggested the implementation of other classes of ships in YPSim, while another two recommended the integration with the BNA's tactical simulator used for other disciplines. There was only one critical review coming from the open-ended questions and it regarded the use of a PC game for a team task trainer. This participant asked the question how YPSim could be used on individual PCs to train in a teamwork situation, considering the OOD, helmsman and lee-helmsman.

C. ANALYSIS

We evaluated the overall results of the survey as extremely positive for the scope of this research. The results showed us important evidence of a user population that has a moderate proportion of gamers and a large proportion of believers in the value of the virtual environment simulation technology. YPs are believed to provide valuable training to the BNA midshipmen. The current learning framework presents significant knowledge gaps, which could be addressed using a ship-driving simulator to increase familiarization and performance aboard.

There is enough evidence to support the relevance of a ship-driving simulator in improving the training experience among the midshipmen population. Using a simulator as an instructional tool could also represent a good contribution to improve midshipmen understanding of such complex topics. By modeling other types of ships that could represent midshipmen's near future as

an officer, the motivational level of the students could increase, representing more knowledge absorption inside the classroom.

The analysis of YPSim's usability brings important insights into how the user population sees this type of technology attached to the game industry. For a generation raised on video games, expectations for graphics and speed are high. Therefore, the relatively moderate acceptance of the interface quality and physics model may be a reflection of the student's history with video games. The results are extremely relevant and indicate that product improvements are needed, regarding the graphics, interface and physical model of the YP. When considered the current version of YPSim as a prototype, made without the help of any professional 3D artist or programmer, we could see a positive user acceptance evaluation.

The user population is motivated over the technology and concept of YPSim, meaning that their minds are opened to use it. The possibility of distributing YPSim for personal use on midshipmen's PCs should be considered as an important step in order to reduce the knowledge gap and increase YP familiarization. Although, this step must be conducted only after the product refinements previously mentioned in order to increase acceptance. Further research must be conducted in order to gather more precise information about the issues affecting presence and usability in precise terms, enabling the specific product improvements.

THIS PAGE INTENTIONALLY LEFT BLANK

VIII. CONCLUSION AND FUTURE WORK

A. CONCLUSION

Using the Brazilian Naval Academy (BNA) environment as a case study scenario, we were able to study the tasks trained aboard naval academy yard patrol (YP) craft. The cognitive process involved in learning navigation and shiphandling material set the correct path towards the development of a game-based simulation tool, intended to reduce the knowledge gap between classroom and aboard instruction. At the end of our research, a proof of concept product, called YPSim, became available for midshipmen use at the BNA at a low cost by using the Delta3D open source simulation game engine.

We presented YPSim to the public at the I/ITSEC 2011 ground floor and received impressive reviews about the concept adopted in the simulator and research results for the version presented. At the BNA, YPSim was originally introduced for study purposes, however, the great acceptance level, from both midshipmen and instructors, made the President to adopt the simulator in the current training framework. Instructors aboard the YPs reported significant improvements in midshipmen performance at sea, after the use of YPSim at BNA as a pre-sail familiarization training. Our project also caught attention form other two full mission bridge simulator projects at the Brazilian Navy, one under development at the University of São Paulo (USP) for the Fleet Training Center, and another by the Center for Naval Systems Analysis (CASNAV) and Fluminense Federal University (UFF) for the Merchant Marine Academy. These two projects present several similarities with YPSim that could be reused in their design and development, representing a practical application of our research.

We proposed the integration of YPSim with a open source research project for Command and Control under development at the Pontifical Catholic University of Rio de Janeiro (PUC-Rio). This work, titled *A Game System Approach for Training and Evaluation: Two sides of the same coin*, became

published in the Proceedings of European Conference on Simulation and AI in Computer Games - GAMEON'11 (Moraes, 2011). In this paper, we presented an integration framework between YPSim and the Command and Control System (CSS) to provide real time feedback of a training exercise at sea to an instructor inside a classroom.

YPSim was tested and refined along these two years of research efforts, now representing a current training tool for the BNA midshipmen. Results of our user acceptance survey were used to support the BNA superintendent’s decision of adopting the current version of YPSim for midshipmen training. We understand that this concept of an easily accessible simulation for ship-driving basic training at naval academies has great potential; although, a future training transfer research is still required. The use of this technology by other institutions, rather than BNA, can be achieved by minor changes in the current design and 3D models.

B. FUTURE WORK

The current version of YPSim was developed only for proof-of-concept purposes and has many limitations before being considered a product that is ready to be used. Future research should be focused on:

- Artificial Intelligence (AI) agents for realistic ship control on the scenario, more specifically for tactical maneuvers missions
- AI agents for realistic representation of the boatswain, helmsman, lee-helmsman and navigator roles
- Intelligent tutoring system using an AI agent
- More refined 3D models
- More refined YP physical mode
- Graphical user interface (GUI) improvements
- Missions tutorials implementation
- Code optimization for better runtime performance using inferior hardware requirements
- HLA/DIS network compatibility
- After Action Review (AAR) capability

The user acceptance survey results indicate that YPSim has a great adoption potential among the midshipmen population, however, a future training transfer study is still necessary to understand the true effectiveness of this tool. We hope that future students find YPSim useful as a research platform in the arena of simulation for training.

Figure 27. BNA’s Superintendent—Rear Admiral Leonardo Puntel, on left—briefing YPSim to a group of retired General Officers visiting the naval academy.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. 3D MODELS

X3D Editor was used only for initial 3D mesh creation of the Brazilian Naval Academy's (BNA) YP model at the very initial stage of the research. Since X3D Editor does not provide support to OSG special nodes functionality, which was required to YPSim, the initial BNA's YP model created in X3D was exported to a file format supported in 3D Studio Max. Later developments in the BNA's YP model were done using 3D Studio Max and the OSG plugins available at the time, allowing OSG node insertion and .osg/.ive exporting features.

The initial option for using Blender, an open source 3D editing software (Salvatore, 2005), did not last long. OSG plugins available for Blender at the time allowed only export/import functionality into OSG supported formats (.osg/.ive), and special nodes features must be implemented using another tool called OSGEdit. These special nodes can be defined as scene graph points accessible by the application allowing special manipulation and control of the 3D model state. YPSim uses the following special nodes in its 3D models:

- Degrees Of Freedom (DOF), allowing runtime manipulations of geometry's rotation inside the model. This node allows, for instance, YPSim to rotate the YP's Radar antenna at any given speed, simply by coding changes in this node's Heading Pitch Roll (HPR) values.
- Switch, allowing runtime manipulation of the displayed geometry for a given part of the 3D model. This node can be used to hold different 3D geometries representing the states of a light bulb on a buoy (on/off). YPSim code can access this node inside the buoy's model and selecting which light bulb to display on the scene.
- Level Of Detail (LOD), allowing runtime selection of which model geometry to be rendered on the scene, based on the distance between the model and scene's camera. This node is mostly used to optimize the number of polygons on the scene, rendering less refined geometries that are too far from the camera.

To facilitate the use of this important OSG functionality in a single editing tool, the author purchased a student license of 3D Studio Max, which was used

in this research project. 3D Studio Max offered more scene management functionality for complex models required in this project, although the same results could be achieved using the open source tools Blender and OSGEdit.

1. YP 3D Model

According to the requirements of YPSim, the geometries on the scene shall provide a fair amount of immersion to the user. The user's YP is the most important 3D mesh present on the scene and must contain a reasonable level of details in order to provide the necessary visual cues to the player's cognitive process during the mission execution. Three different YP 3D models were created on 3D Studio Max for YPSim. Each model represents a real Brazilian Naval Academy YP with hull numbers U10, U11 and U12. The models have the exact same geometries and nodes, but with different textures representing individual names, numbers, call sign, and crest.

Figure 28. The YP U11 being edited in 3D Studio Max – Educational Edition.

The YP 3D model has special features that make it different from other models on YPSim. Several OSG Special nodes were used to allow special runtime manipulations into YP's geometry by YPSim application. These nodes manipulation are important to present a fairly good level of detail and realism required for some of the visual cues needed. Thanks to the DOFTransform OSG node, YPSim is able to access and control rotation parameters of several key geometries inside the YP 3D model, generating nice simulation effects.

The DOFTransform nodes, like any other OSG special node, are not rendered into the scene, keeping it invisible to the user. Special nodes are visible only at the editing tool, and represented by 3D Studio Max as blue 3D axis (Figures 28 and 29). Figure 29 presents the geometries inside the model that are child of a DOFTransform node, allowing the transform manipulations at runtime. Navigation lights can be manipulated using a Switch node, toggling it on/off (Figure 29A). Wind birds for the anemometers can be pointed to the apparent wind and span proportionally to wind speed (Figure 29B). Propeller and rudder movements can be simulated by correspondent DOFTransform parent nodes, providing a nice visualization of the underwater situation (Figure 29D). Inside the bridge, important moving parts will provide valuable visual cues during the maneuvers, such as engine control levers, gyro compass dial, helm, and rudder indicator needle (Figure 29E). Even though YP's .50 caliber machine gun is not covered by the YPSim requirements, DOFTransform nodes were implemented in order to provide a motivational functionality to the youth population of users (Figure 29E).

The YP 3D model also contains DOFTransform nodes for each one of the nine mooring lines on the BNA YPs. These nodes and the corresponding geometries for the mooring lines are not visible in Figures 16 and 17; further details about this feature are given in Chapter III, C.

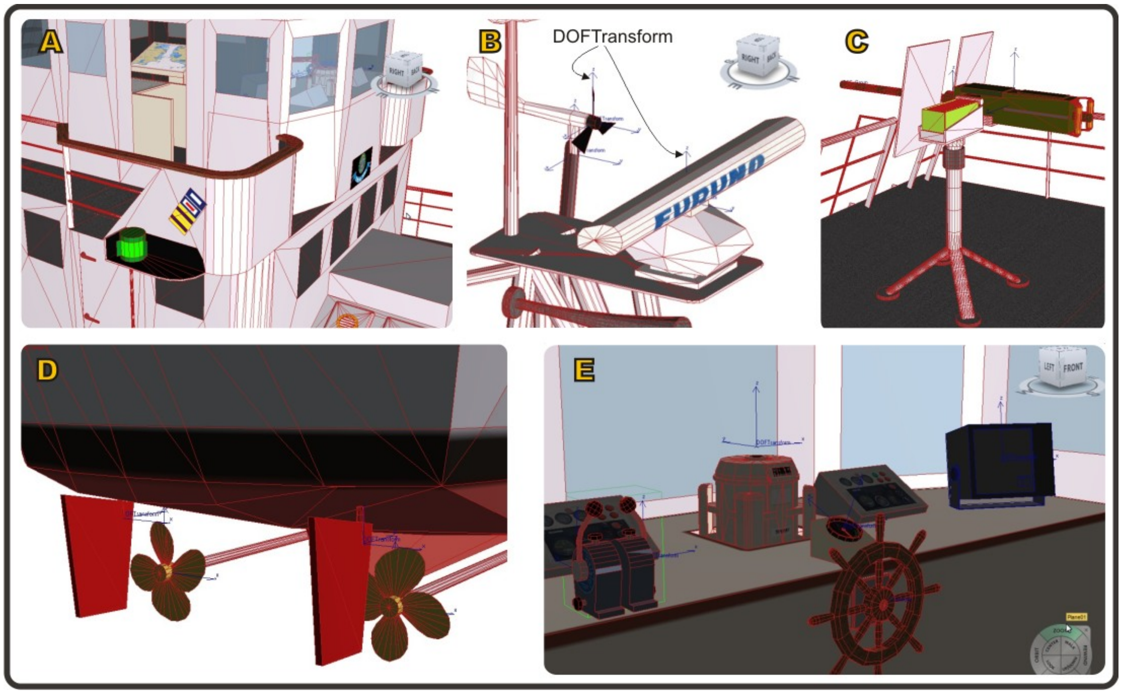


Figure 29. Detailed parts of the YP 3D geometry, containing OSG special nodes.

2. Terrain

YPSim has the Guanabara Bay (Rio de Janeiro – Brazil) as its basic scenario, since this represents the Brazilian Naval Academy's surroundings and most common training environment for the BNA YPs. Due to time constraints and for the scope of this thesis, only three mission environments were developed. The first contains the Brazilian Naval Academy (Guanabara Bay), the second a small set of fictitious islands in the middle of the ocean, and a third one with no terrain, representing Open Ocean.

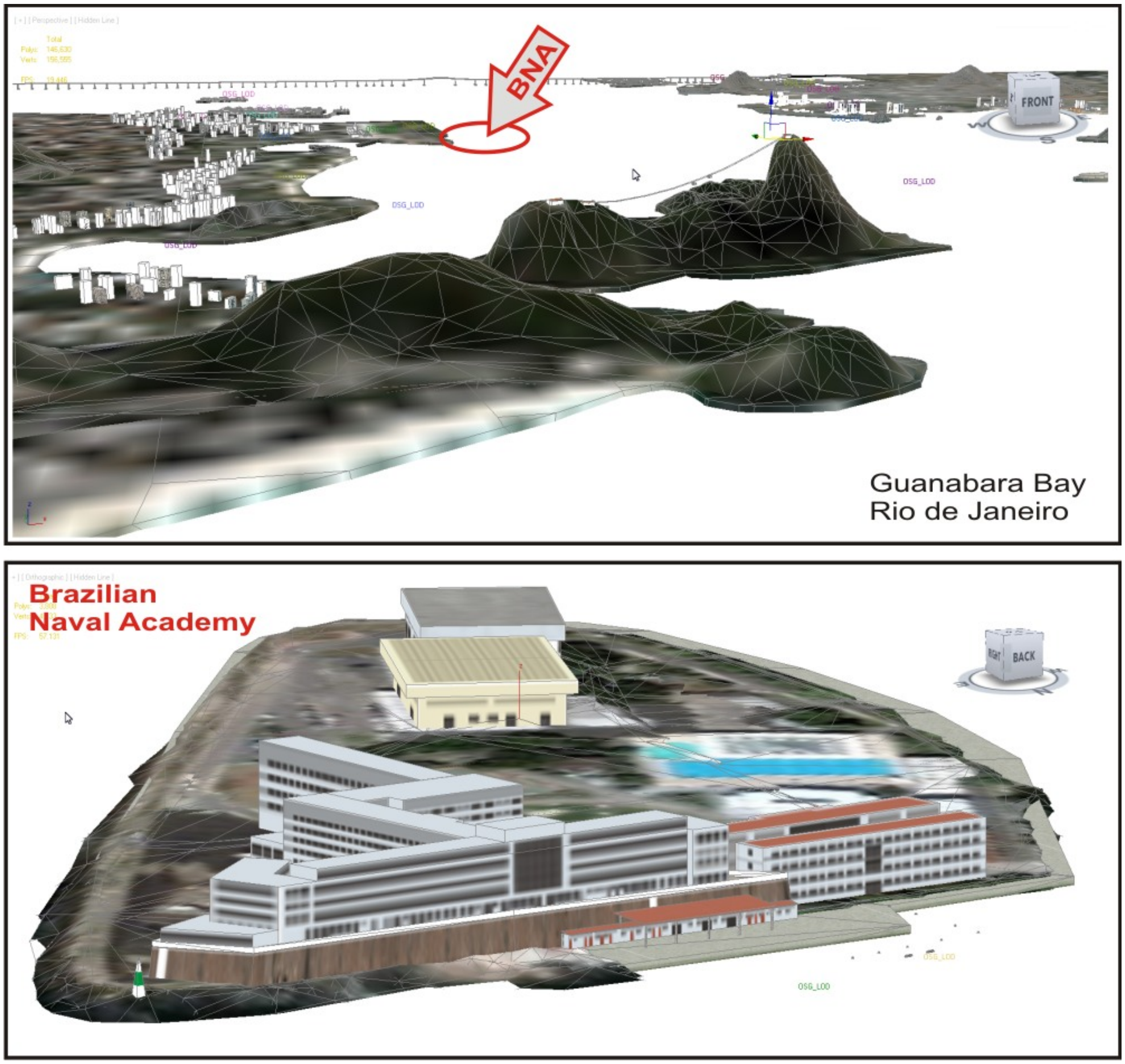


Figure 30. 3D models of the Guanabara Bay (top) and Brazilian Naval Academy (bottom).

The Guanabara Bay terrain was created at the *Laboratório de Sistemas Digitais* (LSI) of the University of São Paulo – Brazil, as part of another virtual environment simulator project under development for the Brazilian Navy (Rodrigues, 2010). The original model was created using Maya editing tool and converted to 3D Studio Max by the author. An extensive editing process took place in order to adapt LSI's original model into the actual YPSim main terrain file. Using Level Of Detail (LOD) nodes and mesh optimization to reduce the number polygons, the model became light enough to run in conventional PCs. A

separated model for the Brazilian Naval Academy buildings was created by the author and added on top of the basic terrain. This step was necessary, given the refinements required for this specific location, used for referencing in most of the maneuvers on YPSim. The most important landmarks and geographic features, such as lighthouses, buildings, hills and coastlines, are represented and correctly positioned in the terrain.

The second terrain file is a simple set of two fictitious islands created by the author in order to provide YPSim with a lighter scenario for better performance, since it has significantly less polygons than Guanabara Bay.

3. Other Scene Objects

In order to have more detailed scenarios, YPSim also uses separate 3D models for navigation buoys, other surface contacts and to represent piers used for mooring.

The navigation buoys use LOD nodes to performance optimization and Switch nodes to control light state (on/off). The geometries used for the navigation buoy follow the standard buoy formats adopted in Brazil, providing a realistic representation of this important navigation aid. A total of 11 types of navigation buoys were created for YPSim and seven of them use LOD nodes in three levels, (a) high definition for distances between 0 and 800 meters, (b) medium resolution, between 800 and 3,000 meters, and (c) low resolution, used at distances greater than 3,000 meters (Figure 31).

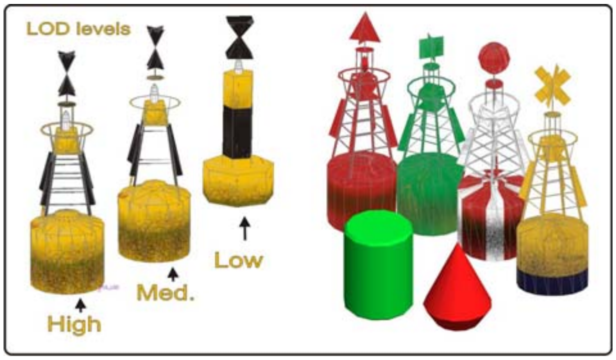


Figure 31. Examples of the 3D models used for buoys.

Surface contacts that compose the scenario use very simple 3D models that contain no OSG special nodes. Figure 32 presents some examples of the models used in YPSim to simulate the typical surface contacts expected at the Guanabara Bay scenario environment. Navy warships transiting to or from the naval base are frequently observed at the BNA's surroundings, bringing an important motivational aspect to the midshipmen aboard the YP (Figure 32A). Merchant ships, fishing and sailing boats represent an important component in the scenario, significantly increasing the Officer of the Deck's (OOD) cognitive processing.

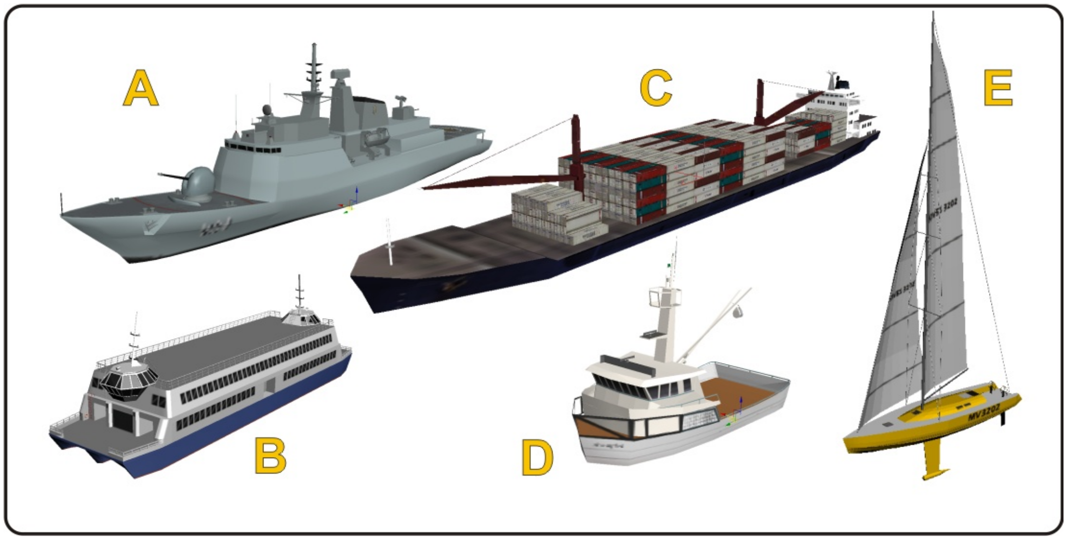


Figure 32. 3D models of the surface contacts used in YPSim.

Piers used for mooring require special OSG nodes (Group) for YPSim access during runtime. Accessing this nodes transform matrices; the code is able to calculate the position of each pier bollard, which is important for mooring. Figure 33 shows the two pier models used on YPSim. The first is fictitious and included at the Islands scenario (A), while the second is an exact representation of the one used at the BNAs (B). The correct position of each bollard and the fenders have extreme relevance for the simulation of a real-world mooring scenario, such as the BNA.

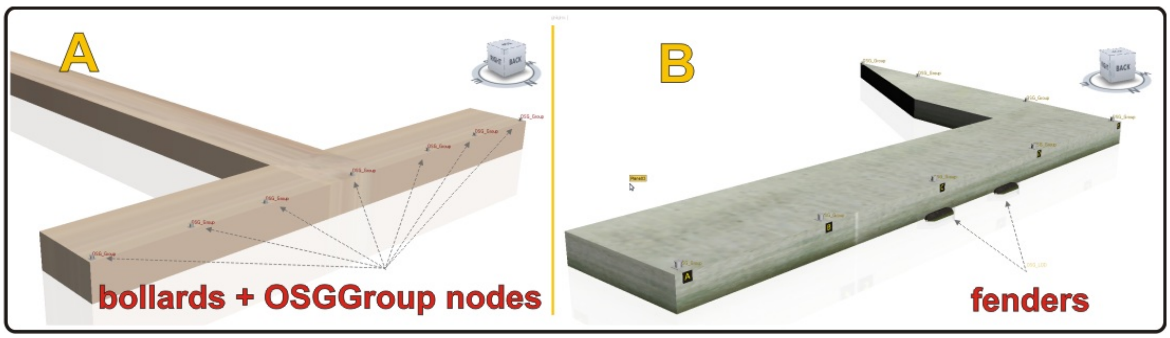


Figure 33. 3D models representing the piers used for mooring.

APPENDIX B. MOORING LINES IMPLEMENTATION

The mooring lines simulation is composed of two parts: visual and physics. The YP has 10 lines available on the deck, one centered at the bow, one centered at the stern and four more each side of the YP. Bollards placed on the deck mark the place where YP's crew handles the mooring line. At the other extreme of each line is another bollard, located at the pier. For each simulated line, a relative force is applied on the YP's body at the corresponding bollard on deck. The relative force vector's direction is given by the pier and deck bollard's alignment, and its magnitude is proportional to the difference of the line length and the distance between bollards, as shown in Figure 34.

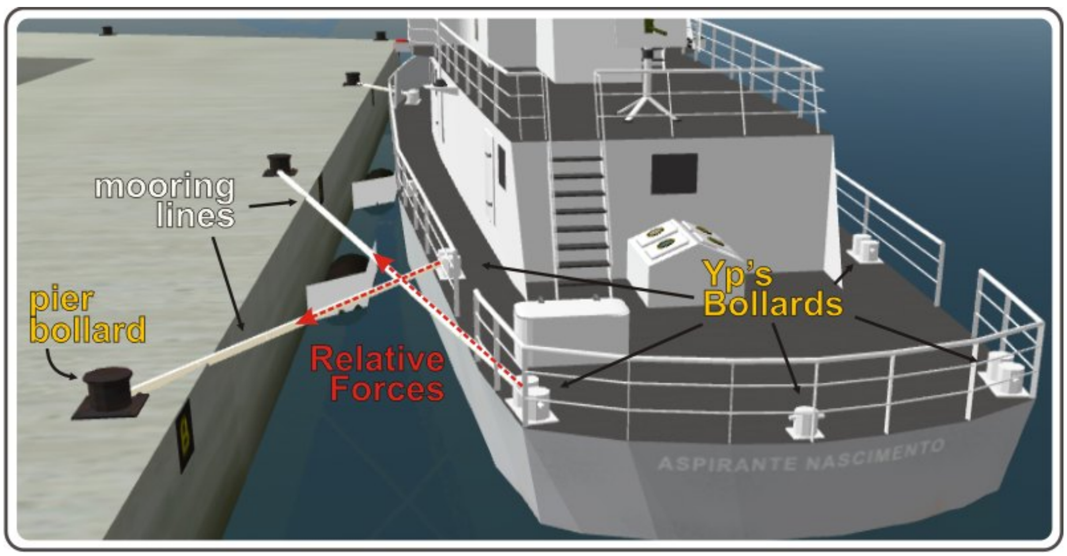


Figure 34. Screenshot of a YPSim mooring maneuver showing the components used in the model

Each mooring line is composed by six cylinders and DOFTransform pairs in a parent child chain. The higher pair in the chain is the one closer to the associated YP's bollard. Each cylinder has one meter in length and is rendered in white in the scene. Cylinders are represented with different colors in Figure 35 for better visualization.

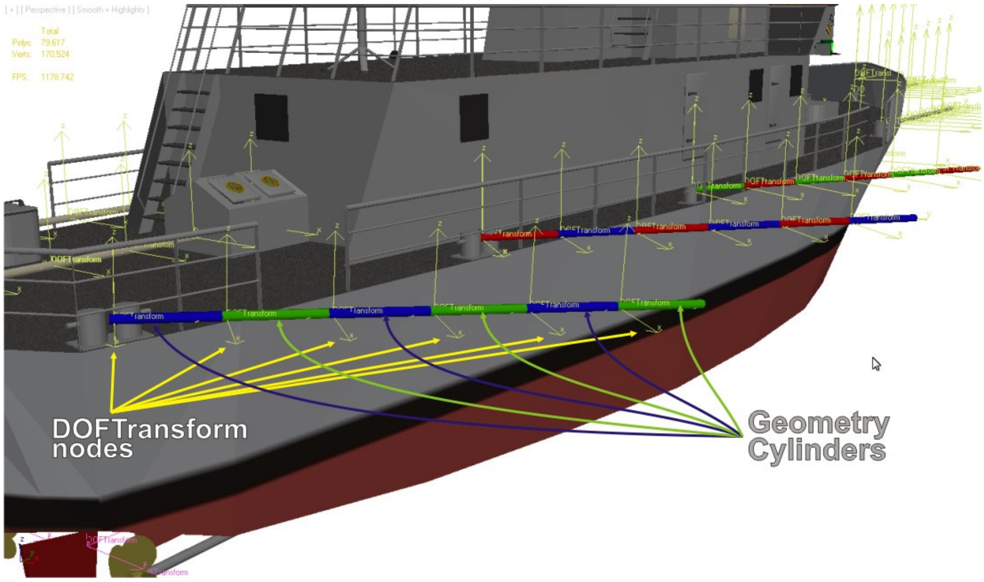


Figure 35. Screenshot of the 3D editing tool showing the DOFTransform nodes and geometries of the mooring lines, incorporated to the YP's 3D model.

In a mooring mission in YPSim, the player can activate or deactivate each line by pressing special keys on the selected input device. The current version of YPSim does not allow the user to select which bollard at the pier to put the line. For the scope of this thesis, each line has its predefined pier bollard position corresponding to the typical mooring at the BNA pier. At the activation, the distance between bollards (pier and YP) is stored, representing the initial line length. Each DOFTransform is rotated towards the pier bollard and scaled to increase cylinder's length in order to match the total line length. A relative spring force is calculated at every physics step, using a vector with direction equivalent to the vector formed between the two bollards. The magnitude of the force will follow a spring model, using a given constant and the square of the difference between the actual inter-bollard distance and the stored line length. If this difference is negative, meaning the line is not under tension, no force is applied.

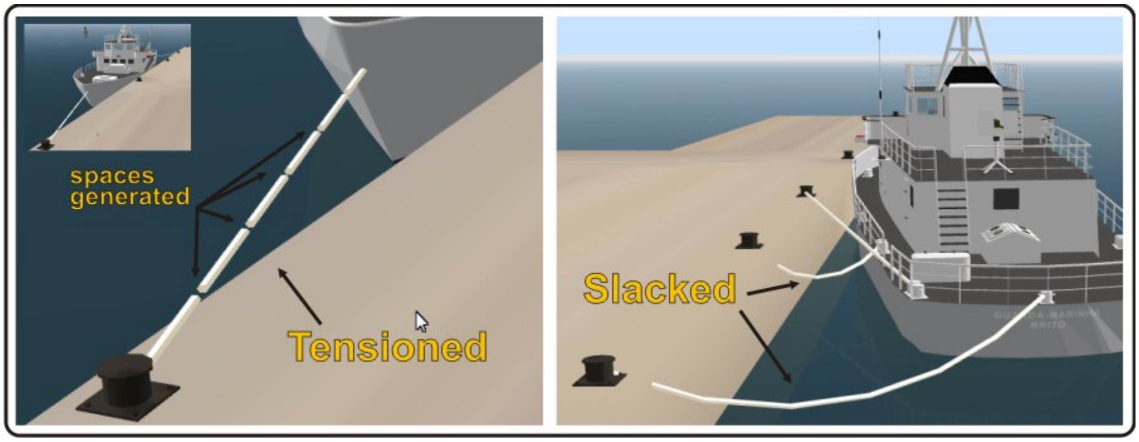


Figure 36. Visualization of the rendering effects simulating tensioned and slacked lines

The DOFTransform nodes for each cylinder are used to scale the cylinders to perfectly match the total line length. If the line is tensioned, separation between lines will appear, providing an important visual cue to the player in order to assess how tight the line is. In case of a slack line, the total length of the line is respected and the rendered effect of an arc is obtained by reorienting each DOFTransform in a systematic way. The arc effect is another important visual cue about the tensioning state of the mooring line. The final rendering effects can be visualized in Figure 36.

Special commands are available to the user, allowing the lines to be thrown off and to heave it, adjusting the stored length to the current inter-bollard distance. For the scope of this thesis, a specific interactive GUI for controlling the mooring lines was not implemented on the current version of YPSim. Methods for improving the mooring line functionality are encouraged in future research.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. ANCHOR IMPLEMENTATION

The anchor model developed in this work is simple and takes into account only the most important factors to simulate the process of dropping and setting the YP's anchor. The model is composed of several objects of the dtCore::Object class, with their physics enabled. The chain is made of several sections and each section is simulated as a separate physical object (dtCore::Object). Chain sections are linked using an ODE ball socket joint (ODE, n.d.), allowing only free angular movement between bodies. The first chain section is attached to the YP's body at the hawse pipe. The anchor is made of two distinct bodies, one representing the anchor shank and the other grouping crown, stock and flukes. The last chain section is connected to the shank using a ball socket joint and the shank is attached to the crown via a hinge joint, with high and low angle limits simulating shank movement restrictions. The lower body of the anchor, composed of crown, stock and flukes, has five contact joints to create friction against the seabed. Figure 37 graphically describes the chain sections and anchor with respective joints. In this case, only six chain sections were used in the model; however, this value is configurable in the scenario mission file. The recommended CPU configuration for YPSim can handle up to 40 sections without significant loss of performance, increasing the refinement of the simulation.

To simulate the complexity of setting the anchor, dynamic values of mass are calculated for the anchor's lower body. In real life, the pulling force applied to the anchor should be as horizontal as possible in order to get more friction from the flukes. If the chain length is not long enough, the pulling force will start to have a vertical component, pulling the anchor upwards and reducing friction. In this model, the friction coefficients applied at each one of the contact points are constant; however, if the anchor is tilted and not horizontally laid on the seabed, fewer contact points will be active, thus reducing friction. The upward pulling effect is simulated by changing the anchor's lower body mass according to the

height of the last chain section attached to the anchor's shank. This measure will be used as an indicator of the direction of the pulling force. When pulling horizontally, the last chain section will be low, close to the seabed. The more vertical the chain pulls the anchor, the higher the last chain section will be, meaning less friction applied at the anchor, simulated by decreasing the mass of the anchor's lower part.

Basic controlling functionality was added to YPSim, allowing the user to drop and heave the anchor, using a chain length, number of sections and friction constant that are previously set at the scenario configuration file. For the scope of this thesis, an interactive GUI was not implemented; however, it is an important component to be added, allowing more control functionality, such as setting the chain length and artificial intelligence to simulate the Boatswain dialogue during the anchoring.

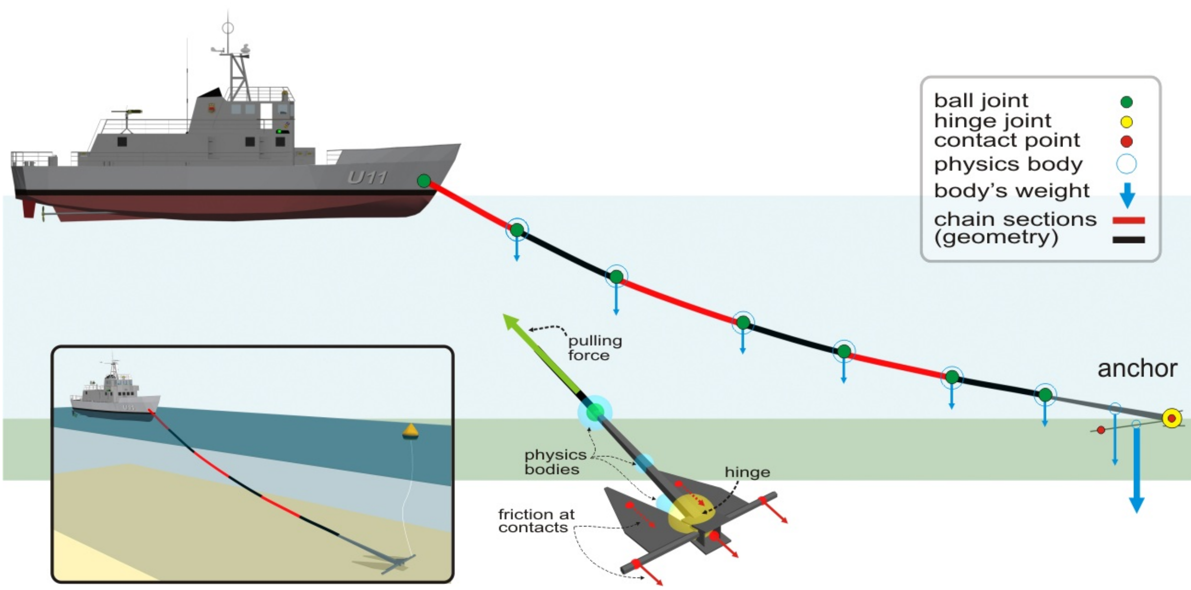


Figure 37. A graphical representation of the anchor and chain model

APPENDIX D. YP PHYSICAL MODEL

The actual YP's physical model has the following forces and torques, better visualized in Figure 38.

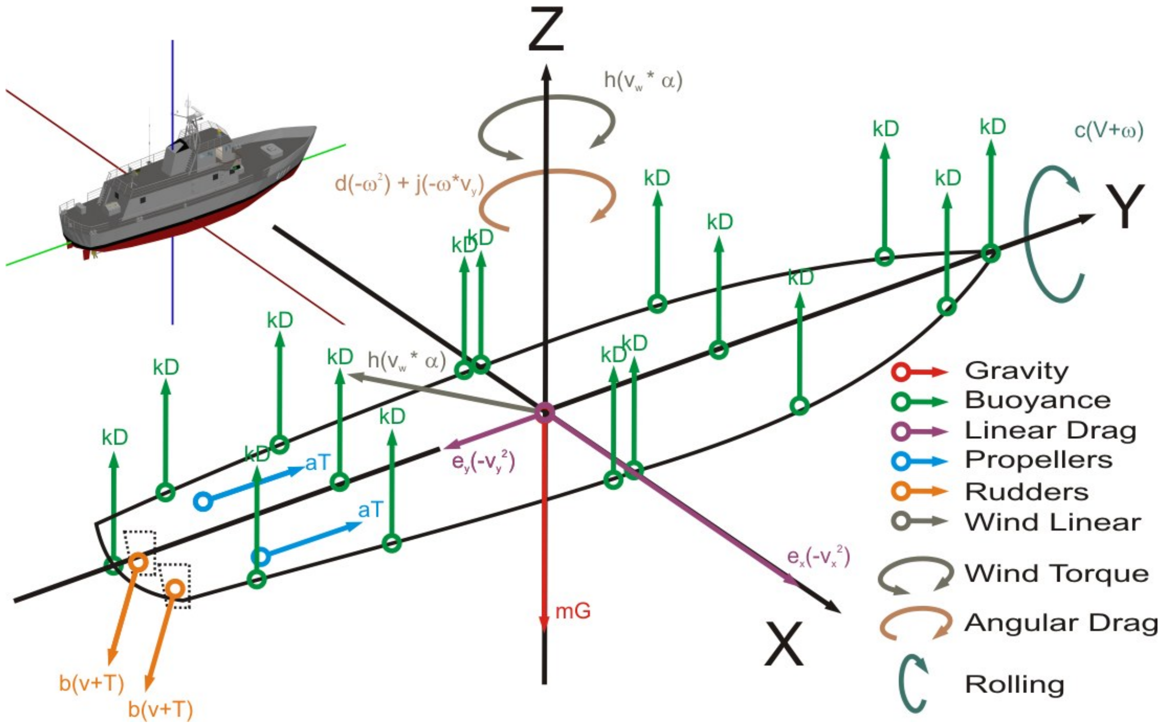


Figure 38. YP's physical model diagram.

- Gravity: applied over the total mass of the YP in its local origin.
- Buoyance: distributed over 16 points over the horizontal plane of the YP. For each point, an ODE contact joint is created with parameters that simulate a floating body, meaning a highly spongy joint. Comparing point's Z coordinate and the ocean surface at point's XY position calculate the point's depth, which is used to set the joint's depth. Each joint is attached to the YP's body, enabling the floating effect over the ocean, with a realistic response to the swell.
- Propellers: two forces are applied independently, one for port and another for the starboard engines. These forces assume a fixed longitudinal direction (YP's Y local axis), moving forward or backwards proportional to the respective engine's RPM. The forces for the forward and backward movements have different parametric

equations, using the engine RPM as an argument. The reason for this difference is in the propeller's profile, designed to be more effective when moving forward. These forces are applied at the position of each propeller, relative to the YP's origin.

- Rudders: two forces are applied, one for each rudder (port and starboard), with direction equals to the rudder's normal. The magnitude of these forces is calculated separately, proportional to the respective water flow through each rudder. The water flow is given not only by the YP's movement over the water but also by the respective propeller when engaged. These are forces applied in local coordinates, at the position of the rudders.
- Linear Drag: a force that is proportional to the square of the YP's velocity over the water and a given pair of friction coefficients (XY local axis). Friction coefficients for the X component of the force (lateral drag) is much higher than Y component (longitudinal drag) as expected because of the hull geometry. Another important effect simulated is the use of different longitudinal components for forward or backward YP movement. The hull offers much less resistance when moving forward than backwards. The velocity vector over the water is calculated subtracting the sea current velocity from the YP's body linear velocity in world coordinates.
- Sea current: the effect caused by the sea current is not result of any direct force or torque being applied at the YP's body. Instead, simply by subtracting the sea current velocity vector from the YP's linear velocity and using the result to calculate the linear drag force, we are able to induce the desired effect. By doing subtraction, the model is always applying dragging forces to the YP's body that are proportional to the sea current velocity.
- Angular Drag: a relative torque applied in the YP's local Z axis, with magnitude proportional to the square of YP's angular velocity in its local Z axis and a rotational drag coefficient. Its direction is the opposite of the YP's rotation in the Z axis, in order to generate the friction effect. Another component is added to the angular drag, given by the ability to a boat straighten out at high speeds. This second component is proportional to the YP's linear velocity and the angular velocity.
- Wind Linear: a force applied at the YP's origin, with direction and magnitude equal to the apparent wind in local coordinates. The force vector is decomposed into XY components and its square multiplied by the respective friction constant. The friction constant is given by the YP's profile over the water, yielding a lateral component greater than longitudinal.

- Wind Torque: a relative torque applied in the YP's local Z axis, with magnitude proportional to the magnitude of the apparent wind and the cube of its angle. The torque direction is given by the sine of the apparent wind angle. The ending result is the simulated natural tendency that the YP has to align to the wind, either facing it or getting it from astern.
- Rolling: in order to simulate the rolling effect that ships present during turns at high speeds, a relative torque in the Y axis was added to the model. The torque magnitude is proportional to the YP's linear velocity and angular velocity in the Z axis. The direction is given by the direction of the turn in progress. When turning to the left, a positive torque in the local Y axis is produced, while right turns make it negative in the same axis.
- Damping: as frequently used in physical simulations, a linear and angular damping is applied to the YP's model, avoiding instability and granting a natural energy dissipation process to the system.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. NAVIGATION RADAR

We implemented the YPSim navigation radar using a raytracing technique to detect contacts and display them on the screen. The Delta3D dtCore::Isector class was used to generate an intersection ray starting at the radar's antenna, pointed to the antenna's Z orientation. The length of the ray is set to the current radar range scale, avoiding detection of off-range contacts. Our algorithm then searches for the first intersection point that occurred, retrieving its XY coordinates in case an intersection happened. This cycle is repeated every simulation frame, updating the ray with a new orientation coming from the antenna's rotation. To generate the radar echo on the screen, we used a Delta3D dtCore::Particle system, with particles being generated at the contact's XY position, now mapped to the radar screen local coordinates. Each particle is set to have a decay time of one antenna's rotation cycle (approximately 2.5 seconds), generating the pleasant effect of a real radar screen.

Figure 39. Screenshot of the YPSim's radar in a HUD mode.

The radar screen can be visualized as a heads-up display or at the radar unit, mounted inside the YP's bridge, according to user's selection. We used a Delta3D dtCore::Object to render the radar screen on the scene graph. The radar's 3D model loaded has special nodes to simulate the radar sweep, electronic bearing line (EBL), variable range marker (VRM) and ship's head marker (Figure 39). These geometry nodes are placed under DOFTransform OSG nodes, allowing selected transform manipulation during the runtime, leading to the desired simulated effect for each one of the nodes. The range scale indicator was implemented using geometries for each digit, controlled by OSG Switch nodes. EBL and VRM values are displayed on the screen by using objects of the Delta3D dtABC::LabelActor class.

Figure 40. YPSim's radar screen displayed on the radar unit, inside the bridge.

APPENDIX F. USER ACCEPTANCE QUESTIONNAIRE

YPSim User Acceptance Survey Part #:

1. Rank (circle one): 1st yr 2nd yr 3rd yr 4th yr **2. YP Mentor (circle one):** Y N

3. Do you play video games frequently (circle one): Y N

4. Have you played any ship simulator game before (circle one): Y N

5. How much time do you need aboard for familiarization until you feel comfortable executing a task for the first time at the YP?

a) Between 10-30 minutes b) Between 30-60 minutes c) > 1 hour d) I did not need any familiarization time

6. In your opinion, what is the best way to learn navigation and ship handling skills aboard the YP?

a) I prefer observe my colleagues doing the tasks and try to extract their lessons from success and fail. I also feel comfortable about asking questions to senior midshipmen and/or the instructors.
b) I prefer observe my colleagues doing the tasks and try to extract their lessons from success and fail. I don't feel so comfortable about asking questions to senior midshipmen and/or the instructors.
c) I prefer doing the tasks by myself and I'm not afraid of doing mistakes, they will be a valuable source of learning for me.
d) I prefer doing the tasks by myself, but I'm afraid of doing mistakes, causing a bad impression to my instructors and/or colleagues.
e) Other (explain):

7. Rate your agreement with the following statements by marking an X in one block for each:

		Disagree				*Agree*
A	I feel confident that my knowledge about the tasks executed aboard will be sufficient.	①	②	③	④	⑤
B	Before going aboard, I have complete understanding of the tasks I'll execute	①	②	③	④	⑤
C	YP hands on training motivates me	①	②	③	④	⑤
D	YP hands on training is important to learn navigation and ship handling skills	①	②	③	④	⑤
E	I feel well prepared every time I go aboard the YP for training	①	②	③	④	⑤
F	There is some knowledge gap between classroom theory and the practice aboard the YP	①	②	③	④	⑤
G	The number of YP training sessions is enough to understand the theory and master the skills of navigation and ship	①	②	③	④	⑤
H	Sometimes I want to execute (or repeat) a maneuver onboard but there is not enough time to do that	①	②	③	④	⑤

Turn page

8. Rate your agreement with the following statements by marking an X in one block for each:

		Disagree				*Agree*
A	The use of a ship driving simulator would help me to reduce any knowledge gap between classroom and the hands on	①	②	③	④	⑤
B	The use of a ship driving simulator would be a motivational tool to academy's midshipmen	①	②	③	④	⑤
C	If I had a ship driving game simulator in which I was able to play the same tasks executed aboard the YP, I would use it to practice before going aboard	①	②	③	④	⑤
D	A YP simulator is not important and will never help me to perform better aboard the YP	①	②	③	④	⑤
E	The use of a YP simulator would be a valuable instructional resource inside the classroom, making it easier to better understand procedures and maneuvers	①	②	③	④	⑤
F	YPSim graphics quality is good	①	②	③	④	⑤
G	YPSim can be a valuable training tool	①	②	③	④	⑤
H	Using YPSim gives me the sensation of being onboard the YP	①	②	③	④	⑤
I	If I had a version of YPSim in my PC I would play it	①	②	③	④	⑤
J	The YP's movements and physics response in YPSim are close to the reality	①	②	③	④	⑤
K	YPSim indicators are easy to read and give the user a complete understand about the YP situation	①	②	③	④	⑤
L	All information needed (controls, sensores, visualization, sound,...) to execute the tasks are available on YPSim	①	②	③	④	⑤
M	The use of YPSim can motivate midshipmen and make them feel closer to the future job aboard a real ship	①	②	③	④	⑤
N	Using YPSim will be a waste of time, no simulation can replace the hands on training aboard the YP	①	②	③	④	⑤
O	YPSim has poor quality and is too distant from a good simulators that could be used as a valid training tool	①	②	③	④	⑤

9. Do you have any suggestions for improving YPSim as a training tool?

10. Provide any comments about your general impression of YPSim and ideas about situations where this simulation would be useful to train or provide learning to you. Please, consider the possibility of using it as a classroom simulator, instructional resource or a PC-game.

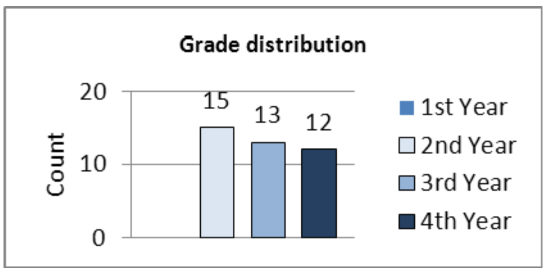
Thank you!!!

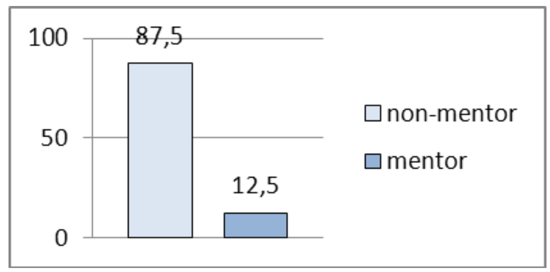
APPENDIX G. SURVEY RESULTS

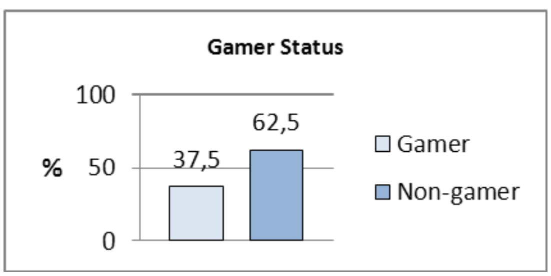
The distribution of the answers obtained from the forty participants on the user acceptance survey is graphically represented in Table 14. For the Likert scale questions, 1 means fully disagrees while 5 means fully agrees with the proposed statement. Open-ended questions were not summarized here.

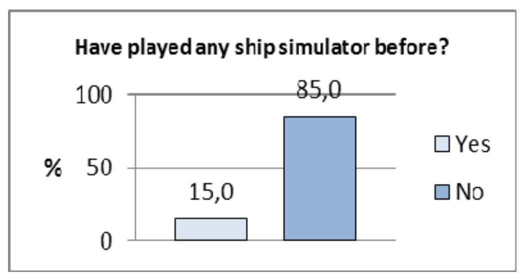
Table 12. Graphical representation of the user acceptance questionnaire answers.

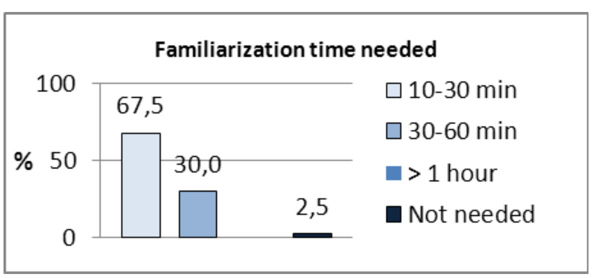
Answers' Distributions

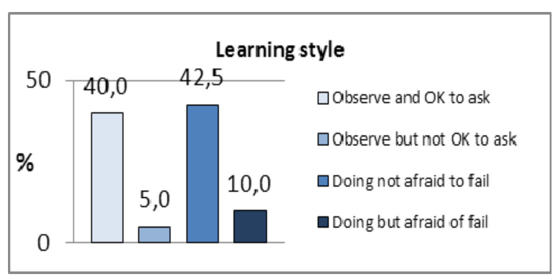


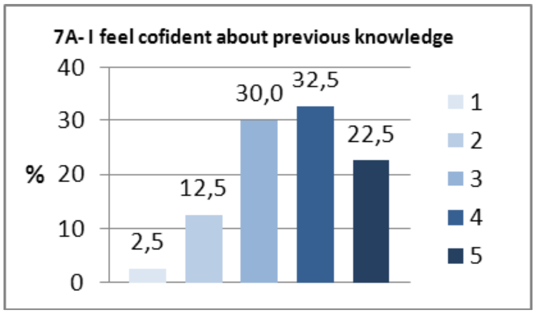


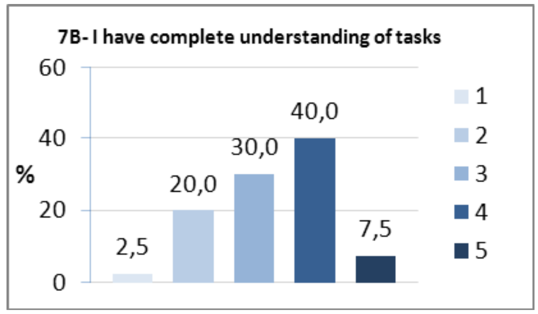


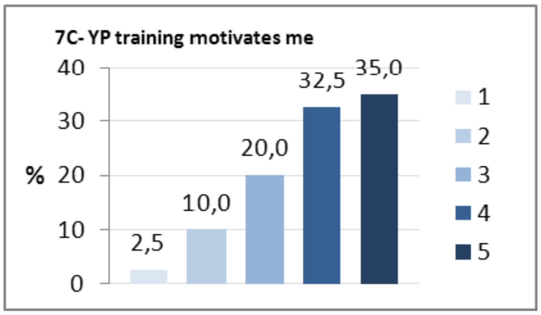


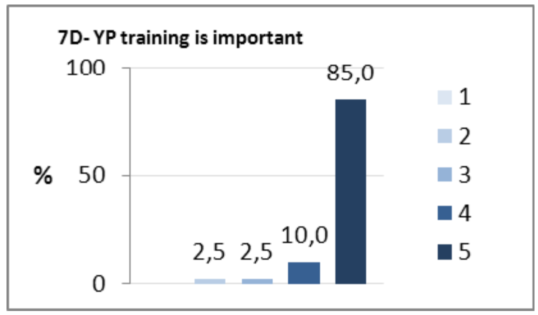


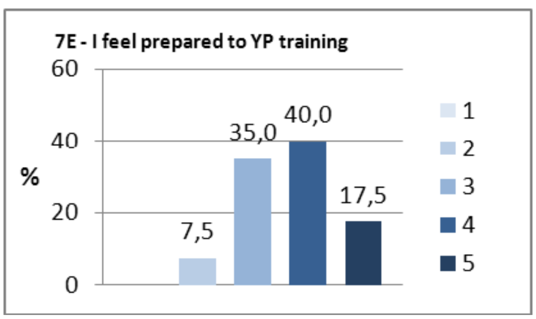


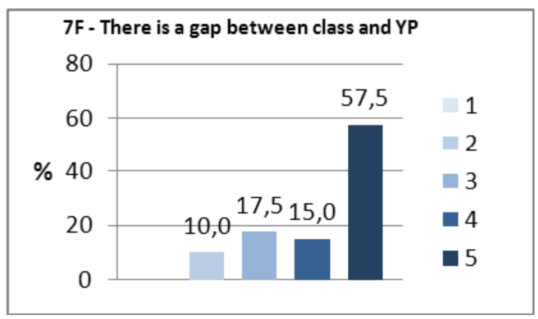

7A- I feel cofident about previous knowledge
%
0
10
20
30
40
2,5
12,5
30,0
32,5
22,5
1
2
3
4
5

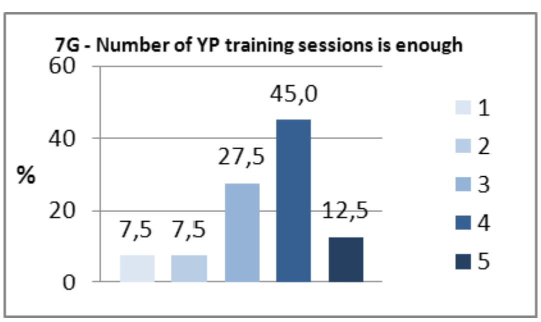

7B- I have complete understanding of tasks
%
0
20
40
60
2,5
20,0
30,0
40,0
7,5
1
2
3
4
5

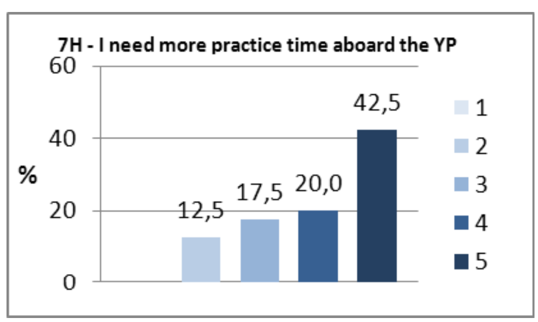

7C- YP training motivates me
%
0
10
20
30
40
2,5
10,0
20,0
32,5
35,0
1
2
3
4
5

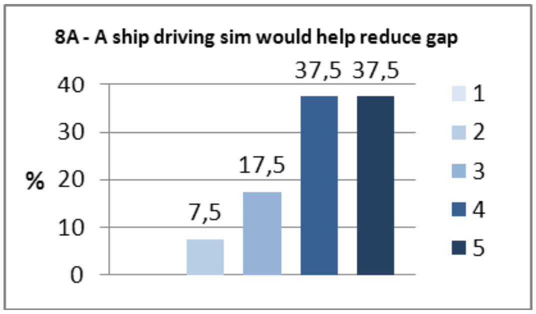

7D- YP training is important
%
0
50
100
2,5
2,5
10,0
85,0
1
2
3
4
5

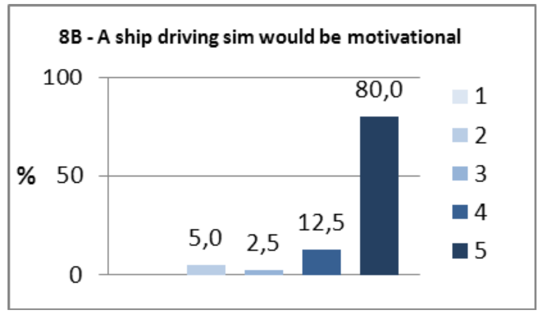

7E - I feel prepared to YP training
%
0
20
40
60
7,5
35,0
40,0
17,5
1
2
3
4
5

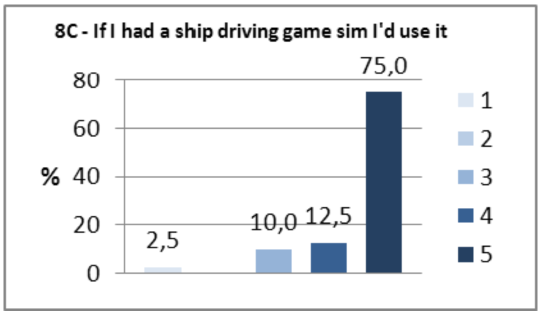

7F - There is a gap between class and YP
%
0
20
40
60
80
10,0
17,5
15,0
57,5
1
2
3
4
5

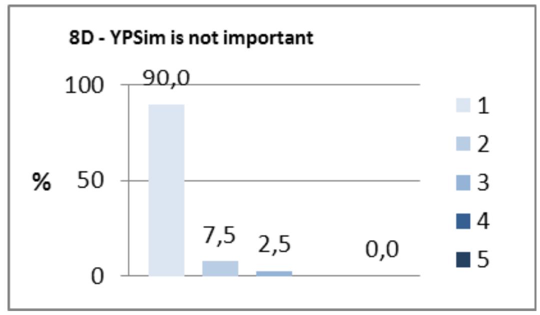

7G - Number of YP training sessions is enough
%
0
20
40
60
7,5
7,5
27,5
45,0
12,5
1
2
3
4
5

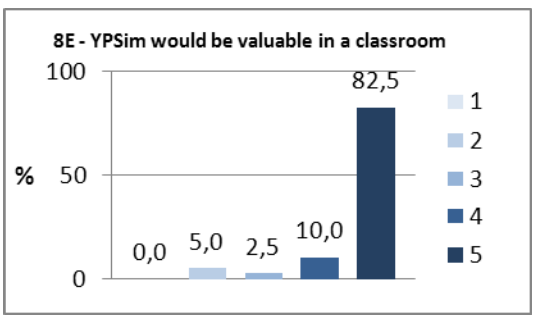

7H - I need more practice time aboard the YP
%
0
20
40
60
12,5
17,5
20,0
42,5
1
2
3
4
5

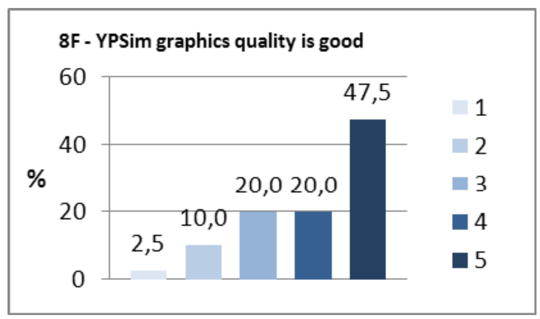

8A - A ship driving sim would help reduce gap
%
40
30
20
10
0
7,5
17,5
37,5
37,5
1
2
3
4
5

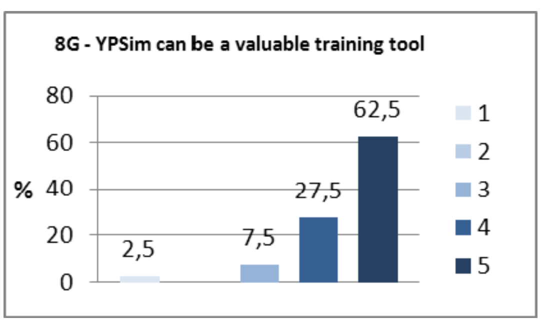

8B - A ship driving sim would be motivational
%
100
50
0
5,0
2,5
12,5
80,0
1
2
3
4
5

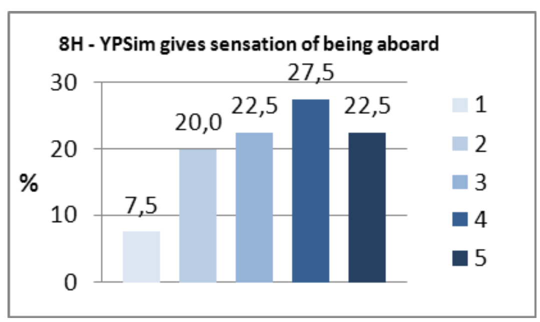

8C - If I had a ship driving game sim I'd use it
%
80
60
40
20
0
2,5
10,0
12,5
75,0
1
2
3
4
5

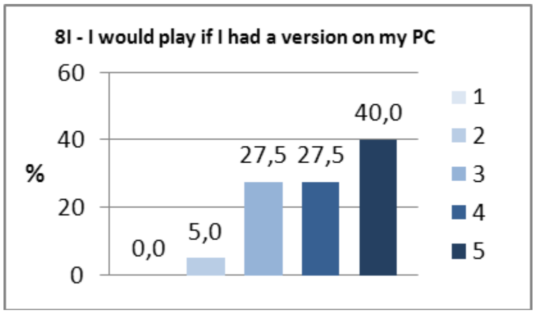

8D - YPSim is not important
%
100
50
0
90,0
7,5
2,5
0,0
1
2
3
4
5

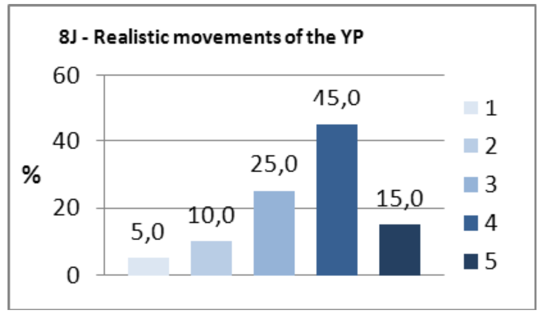

8E - YPSim would be valuable in a classroom
%
100
50
0
0,0
5,0
2,5
10,0
82,5
1
2
3
4
5

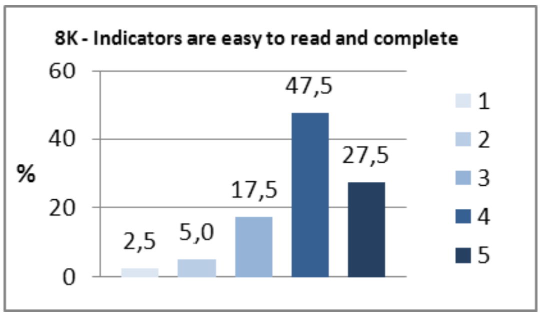

8F - YPSim graphics quality is good
%
60
40
20
0
2,5
10,0
20,0
20,0
47,5
1
2
3
4
5

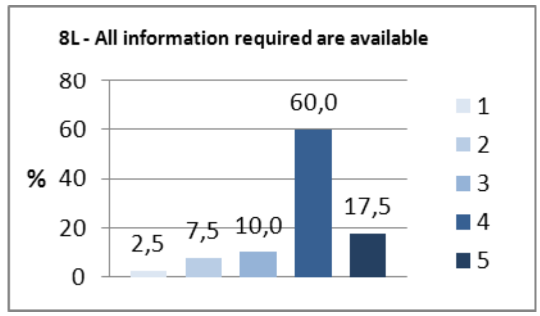

8G - YPSim can be a valuable training tool
%
80
60
40
20
0
2,5
7,5
27,5
62,5
1
2
3
4
5

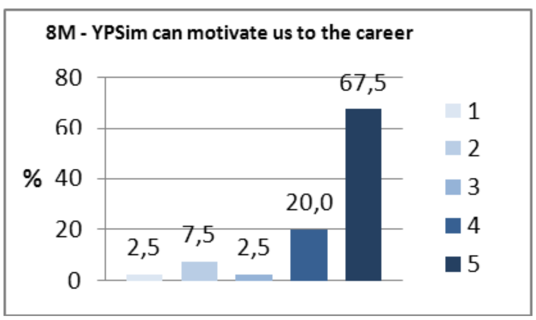

8H - YPSim gives sensation of being aboard
%
30
20
10
0
7,5
20,0
22,5
27,5
22,5
1
2
3
4
5

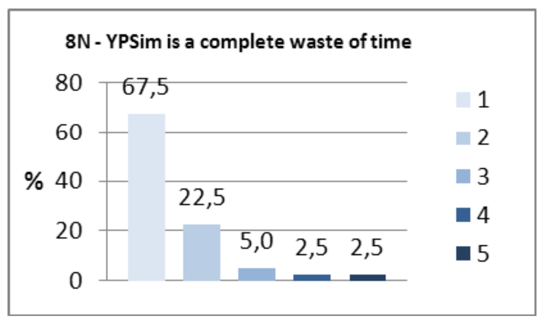

8I - I would play if I had a version on my PC
%
60
40
20
0
0,0
5,0
27,5
27,5
40,0
1
2
3
4
5

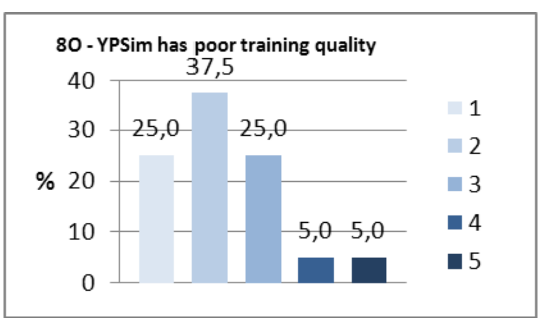

8J - Realistic movements of the YP
%
60
40
20
0
5,0
10,0
25,0
45,0
15,0
1
2
3
4
5


8K - Indicators are easy to read and complete
%
60
40
20
0
2,5
5,0
17,5
47,5
27,5
1
2
3
4
5


8L - All information required are available
%
80
60
40
20
0
2,5
7,5
10,0
60,0
17,5
1
2
3
4
5


8M - YPSim can motivate us to the career
%
80
60
40
20
0
2,5
7,5
2,5
20,0
67,5
1
2
3
4
5


8N - YPSim is a complete waste of time
%
80
60
40
20
0
67,5
22,5
5,0
2,5
2,5
1
2
3
4
5


8O - YPSim has poor training quality
%
40
30
20
10
0
25,0
37,5
25,0
5,0
5,0
1
2
3
4
5

APPENDIX H. HIERARCHICAL TASK ANALYSIS (MAN OVERBOARD TASK)

Table 13. Hierarchical Task Analysis for the Man Overboard task.

Step	Function	User Action/Task	Remarks
1	**Man overboard recovery (Anderson Turn)**		
1.1	Execute Maneuver	Conduct the MOB maneuver according to procedures in shiphandling manual for a MOB recovery during day time.	
1.1.1	Check surroundings for any contact or hazard	OOD goes to the bridge wing of the side of turn and visually check for contacts or navigation hazards.	This evaluation should be done quickly in order to start turning in short time and abbreviate the maneuver.
1.1.2	Move the ship's stern away from the MOB	Order the Helmsman full turn to the same side the MOB felt.	Helmsman acknowledges the order. Rudder indicator goes to full rudder angle at the side desired. Cues from the horizon movement trigger the sensation of moving to the correct direction. Fill the ship rolling to the correct side, another good indicator of the turning.
1.1.3	Increase ship's speed	Order "ALL ENGINES AHEAD FLANK" to the Lee Helmsman.	Lee Helmsman acknowledges the order. Ship's speed indicator starts to increase. Acoustical cues from the engines sound changing indicate acceleration. Visual cues from the ship's wake increasing also helps.
1.1.4	Reduce ship's speed	When MOB is at abeam position, order "ALL ENGINES AHEAD 2/3" to the Lee Helmsman.	Lee Helmsman acknowledges the order. Ship's speed indicator starts to decrease. Acoustical cues from the engines sound changing indicate deceleration. Visual cues from the ship's wake changing helps.

Step	Function	User Action/Task	Remarks
1.1.5	Reduce ship's speed again	When MOB is at your port/stbd-bow position, order "ALL ENGINES AHEAD 1/3" to the Lee Helmsman.	The same than 1.1.4.
1.1.6	Reduce turning rate	Right after step 1.1.5, order the Helmsman "EASY YOUR RUDDER."	Helmsman acknowledges the order. At this time, OOD will probably be at the bridge's wing, with a clear view of the MOB, and have no more access to the ship's indicators. OOD must rely on information provided by his/her teammates. A very good visual cue is the rate that the MOB approximates to ship's bow.
1.1.7	Steady the course and approach the MOB	Order the Helmsman "RUDDER AMIDSHIPS."	The same than before, from this point OOD makes small corrections to keep the MOB at ship's bow, slightly to the recovery side (port/stbd.).
1.1.8	Break ship's inertia	When MOB is close to ship's bow, order "ALL ENGINES STOP" to Lee Helmsman.	OOD feels ship's speed being reduced to zero, the deceleration rate needs to be such that the ship is stopped when the MOB is at the recovering position. This is very hard to achieve without reversing the engines to make a good stop.
1.1.9	Make the final approach to the recovery position	Assuming the rescue swimmer method for recovery, maneuver the ship with rudder and engines to place the MOB close to 15 yards from the recovery station. The recovery station is located at the forecastle, at the side chose by OOD to pick-up MOB.	The visual cues will be crucial at this point, doing a good maneuver depends upon OOD's reaction time to the cues present. Quick reactions tend to lead in maneuvers that are more precise.
1.1.10	Recover the man	Determine the Boatswain to recover the MOB	Boatswain acknowledges the order. OOD must watch for not using engines during the pick-up, avoiding risk of hit MOB with the propeller.
1.2	Check MOB bearing and range and ship's state	Evaluate the relative bearing of the MOB and ship's movement. Decide if the situation matches the procedural conditions to act changing ship's course	Collect information from own observation, GPS or Quartermaster of the watch (QOA), plus ship's indicators for rudder and engine. Mental evaluation for changes in rudder and engines orders. This step represents a continuous

Step	Function	User Action/Task	Remarks
		or speed.	processes throughout the maneuver.
1.3	Check environmental conditions	Determine Junior OOD (JOOD) to read water temperature register, time of MOB and calculate direction and magnitude of the wind.	JOOD acknowledges the order and report the information required.
1.3.1	Determine the recovery side	Check the sea state, wind conditions and MOB consciousness to decide which side to approach the MOB (port/stbd.). Inform Boatswain the recovery side.	The visual cues are very important in this action, it is important to be able to check the wind direction and sea state just by looking at the caps of sea waves. During the recovery, OOD must be focused in keeping MOB at leeward, since the wind effect over the ship tend to push it towards the MOB in this case.
1.4	Execute administrative procedures	Start a set of actions that are not correlated with the maneuver cognitive processing.	The sequence is not mandatory, allowing changes between 1.4.1, 1.4.2 and 1.4.3. Step 1.4.4 comes prior to 1.4.5, but could be executed prior to the any other in this group.
1.4.1	Plot MOB position on chart	Give order to plot the MOB position on nautical chart and GPS.	Quartermaster of the Watch is responsible for plotting the MOB position.
1.4.2	Make six short blasts	Give order to make six short blasts on ship's whistle.	Lee-helmsman is responsible for making the blasts.
1.4.3	Drop lifebuoy and smoke sign (assuming day light situation)	Give order to drop a lifebuoy and a smoke sign close to the MOB's position.	Junior OOD (JOOD), the OOD assistant, is responsible for dropping lifebuoy and smoke sign.
1.4.4	Hoist OSCAR flag	Give order to hoist OSCAR flag.	Signal Station is responsible for hosting the OSCAR flag.
1.4.5	Disseminate the MOB situation	Determine JOOD to disseminate "Ship's MOB" in the Power Amplified (PA) system.	JOOD acknowledges the order and executes it.
1.4.6	Dismiss stations.	MOB is recovered, MOB determines all stations to resume normal activities.	Resume ships route, course and speed.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX I. HIERARCHICAL TASK ANALYSIS (ANCHORING)

Table 14. Hierarchical Task Analysis for the Anchoring task.

Step	Function	User Action/Task	Remarks
2	**Anchoring**		
2.1	Pre-action Procedures.	Acquire all important information available about maneuver (1 hour prior to scheduled start).	Conduct a “hot-briefing” at the Bridge with the Boatswain and Navigator.
2.1.1	Check anchoring position in the nautical chart.	Ask Navigator to show and describe the anchorage in the nautical chart.	Conduct a mental evaluation of the proposed plan to the Anchoring maneuver, check safety and effectiveness of the route to anchorage.
2.1.2	Check the anchoring depth and nature of the seabed.	Ask Navigator to show the estimated depth and the expected nature of seabed.	Quickly evaluate the data provided, checking about safety limits and ships draught. Calculate the necessary scope of chain and inform Boatswain.
2.1.4	Check expected tide level and current.	Ask Navigator about the expected tide current at the anchorage.	Make a quick evaluation of how these factors will affect the maneuver.
2.1.3	Check meteorological conditions.	Check the last records of wind direction and speed and expected forecast for the next hours.	Make a quick evaluation of how these factors will affect the maneuver. The final approach to the anchorage should be facing the prevailing factor (wind or current).
2.1.4	Disseminate OOD intentions.	OOD tell Navigator and Boatswain his/her maneuver intentions.	It is very important to make it clear so everyone is able to foresee any potential safety issue.
2.2	Navigate to the anchoring position and set anchor.	Conduct the ship to the anchorage giving orders to the Helmsman and Lee Helmsman, based upon the Navigator’s suggestions. During the whole period of navigation, OOD reports range to anchorage (2000, 1000, 500, 200, 100, 50 yards) to the Boatswain. When ship reaches the anchoring position, properly set anchor.	Navigator provides a complete report each 3 minutes to OOD, so he/she can correct course and speed to destination. Boatswain acknowledges the range to anchorage information, when disseminated by OOD. OOD uses navigation radar and binoculars to check the surrounding environment for surface contacts or potential navigation hazards.

Step	Function	User Action/Task	Remarks
2.2.1	Evaluate ship's status and decide	OOD creates a mental picture of the ship's status and decides whether to apply corrections on present course and/or speed.	The OOD is responsible to bring the ship to the anchorage point following a given route. When ETA (Estimated Time of Arrival) is required to get there, OOD must consider eventual changes in speed, if not he/she will apply changes on course in order to compensate drifting effects from wind and/or current. The OOD relies mainly on information coming from the Navigator to make his/her decisions until reaching the anchorage. OOD's perception of the environment will help by visual observations, assisted by binoculars, and navigation radar. Boatswain information will provide guidance after dropping the anchor. There are three standard speed reductions conditioned to the distance to the anchorage.
2.2.2	Adjust course and speed (when needed)	OOD gives new orders to Helmsman and Lee Helmsman, for course to steer and engines RPM respectively.	If OOD evaluates that a change on course and/or speed is required, he/she will apply it. These changes can be made to increase safety (avoiding any hazard situation) or keep ship on navigation route and schedule.
2.2.3	First speed reduction.	At 1,000 yards from the anchoring position, reduce ships speed to 6 knots, if navigating at greater speed. Determine to the Lee Helmsman "ALL ENGINES AHEAD 2/3."	Lee Helmsman acknowledges the order. Reduction on ship's wake is a good indicator of speed reduction. Engine sound also changes, and speedometer drops to 6 knots.
2.2.4	Second speed reduction.	At 300 yards from anchorage, OOD reduces ship's speed to 3 knots. Determine to the Lee Helmsman "ALL ENGINES AHEAD 1/3."	The same than before, but now speedometer drops to 3 knots.
2.2.5	Third speed reduction	At 100 yards from the anchoring position, OOD stops all engines. Determine to the Lee Helmsman "ALL ENGINES STOP." Disseminate to the Boatswain "STAND BY TO DROP ANCHOR."	The same than before, but now ship is expected to gradually stop. Boatswain acknowledges the order from the forecastle.
2.2.6	Break ship's inertia.	At 50 yards from the anchoring position, break the inertia by reversing all engines. Determine to the Lee Helmsman "ALL ENGINES ASTERN 1/3."	Visual cues coming from the water movement close to the hull are fundamental to OOD perception of ship's stop. Also, if ship is close to shore, OOD can align two ashore objects at abeam and check their relative motions to assess ship's longitudinal

Step	Function	User Action/Task	Remarks
			movement.
2.2.7	Drop Anchor.	Determine Boatswain to drop anchor. Determine Lee Helmsman "ALL ENGINES STOP."	When OOD feel the ship moving astern, by his/her visual cues, around 1 knot, drop anchor is ordered to Boatswain. Boatswain acknowledges the order. Listen the anchor drop on the water and chain unreeling from windlass.
2..2.8	Set the anchor.	Slightly pull the anchor by moving the ship astern at very low speed.	Providing some astern engine power will create a tension in the chain and this will allow the anchor to grip the seabed, holding the ship. Observe the angle that the chain is entering the water, this is a very good indicator if the chain has enough tension or not. If the anchor is not properly gripped at the seabed, the ship will keep moving astern. OOD uses visual cues (water movement or ashore objects) to check ship's movement. Boatswain frequently reports chain status (tension and direction) and finally if the anchor is properly set to the seabed.
2.3	Receive navigation report	Listen to a set of standardized information concerning ship's position relative to the navigation route.	This report is periodically (usually each 3 minutes) issued by the Navigator and verbally acknowledge by the OOD.
2.4	Check environment	Observe the outside environment seeking for relevant information concerning the navigation. Check navigation radar for contacts or any other navigational information.	This environmental check is executed at every moment OOD looks outside, and more consistently right after step 2.3. This check of the surrounding factors is important to support OOD's decisions on changing course, speed and evaluating ship's safety condition (avoiding other contacts).
2.5	Receive Boatswain's report and check Anchor Buoy	Listen to a set of standardized information concerning the anchor and chain status. Visually check the Anchor Buoy, whenever clear, for estimating anchor's position relative to the ship and possible drag.	This information will guide OOD's decisions at the final stage of the navigation to the anchorage (when Boatswain reports ready to drop anchor) and after dropping the anchor. From the moment the anchor is dropped, OOD will rely on Boatswain's report to create a mental picture of what is happening underwater. Information about the anchor conditions (holding or dragging, how the anchor is tending - "9 o'clock," "12 o'clock," number of shots of chain, etc.), and what sort of tension is on the anchor chain will be vital to OOD set the

Step	Function	User Action/Task	Remarks
			anchor (USNA, 1991).
2.6	Disseminate "Anchor Stations"	Determine the Junior OOD (JOOD) to disseminate "ANCHOR STATIONS" in the PA System. (usually five Nautical Miles from anchorage).	JOOD acknowledges the order and disseminates information. All the Stations report when "ready." Once all the stations (Navigator and Boatswain) report ready, OOD is under satisfactory conditions to anchor the ship.
2.6.1	Dismiss stations	The OOD determine one long blast on ship's whistle, "shift colors," display anchoring shape/lights, set anchor watch and dismiss all stations from "Anchor Stations."	Happens after OOD certification that ship is not underway anymore (primarily provided by Navigator). OOD hears the long blast, watch national ensign and union jack hoisted on the flagstaff and jackstaff, respectively, and watches the anchoring shape/light displayed.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Barber, J. A. (2005). *Naval shiphandler's guide*. Annapolis, Maryland: Naval Institute Press.

Beard, C., Wilson, J. P. (2006). *Experiential learning : A Handbook of Best Practice for Educators and Trainers.* London, GBR: Kogan Page Ltd., 144.

Brannon, D. A., Villandre, M. (2002). The forward observer personal computer simulator (FOPCSIM). Master's thesis, Naval Postgraduate School, Monterey, CA.

Bowditch, N. (2002). *The American practical navigator*. Bethesda, Maryland: National Imagery and Mapping Agency.

Caird, J. K. (1996). Persistent Issues in the Application of Virtual Environment Systems to Training. *Proceedings of the 3rd Symposium on Human Interaction with Complex Systems (HICS '96).* IEEE Computer Society, Washington, DC, 124.

Campos, D.V.; Seixas, R.B.; Command and Control: A low cost framework to remotely monitor military training, Proceedings of Spring Simulation Multiconference - SpringSim'11, in Military Modeling and Simulation Symposium - MMS, Boston, MA., 2011.

Cardoso, A. et al (2003). Ambientes Virtuais – Projeto e Implementação. SBC. ISBN 85-88442-67-1.

Clark, R. E., Feldon, D., Van Merrienboer, J. J. G., Yates, K., and Early, S. (2008) Cognitive task analysis. In J. M. Spector, M. D. Merrill, J. J. G. van Merrienboer, & M.P. Driscoll (Eds.). *Handbook of research on educational communications and technology (3rd ed.)*. Mahwah, NJ: Lawrence Erlbaum Associates.

Crandall, B., Klein, G., Hoffman, R. R. (2006). *Working Minds : A Practitioner's Guide to Cognitive Task Analysis.* Cambridge, Massachusetts: MIT Press, 141.

Crenshaw, R. S. (1975). *Naval shiphandling*. Annapolis, Maryland: Naval Institute Press.

DAU (2001). *Systems Engineering Fundamentals.* Fort Belvoir, Virginia: Defense Acquisition University Press.

Ernst, R. B. (2006). The Design of a Stand-Alone Division Tactics Simulator Utilizing Non-Proprietary (Open Source) Media and Iterative Development. (Master's Thesis, Naval Postgraduate School, Monterey, CA).

Fleetman Desktop. (n.d.). Retrieved May 13, 2011, from http://www.dtmedia.co.uk/fleetman.htm

Grassi, C. (2000). A Task Analysis of Pier Side Ship-Handling for Virtual Environment Ship-Handling Simulator Scenario Development. (Master's Thesis, Naval Postgraduate School, Monterey, CA).

Hackos, J. T., Redish, J. C. (1998). *User and Task Analysis for Interface Design*. New York, New York: John Wiley & Sons, Inc., 41.

Hooyer, H. H. (2004). *Behavior and Handling of Ships*. Centreville, Maryland: Cornell Maritime Press, Inc..

Hutchins, Edwin (1995). *Cognition in the wild*. Cambridge, Massachusetts: The MIT Press.

Klein, G.A., Calderwood, R., MacGregor, D. (1989). "Critical decision method for eliciting knowledge," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 3, 462–472, May/June 1989

McDonough, J. & Strom, M. (2005). The Forward Observer Personal Computer SIMulator (FOPCSIM) 2. (Master's thesis, Naval Postgraduate School, Monterey, CA).

Moraes, C.C.; Campos, D.C.; Seixas, R.B.; Day, M.A.; A Game System Approach for Training and Evaluation: Two sides of the same coin, Proceedings of European Conference on Simulation and AI in Computer Games - GAMEON'11, Galway, Ireland, 2011.

Miguens, A. P. (n. d.). *Navegação Ciência e Arte Vol.*(Navigation Science and Art). Retrieved May 13, 2011, from https://www.mar.mil.br/dhn/bhmn/publica_manualnav.html

Neisser, U. (1976). *Cognition and reality.* New York, New York: W. H. Freeman and Company.

Norris, S. D. (1998). A Task Analysis of Underway Replenishment For Virtual Environment Ship-Handling Simulator Scenario Development. (Master's Thesis, Naval Postgraduate School, Monterey, CA).

Open Dynamics Engine (ODE). (n.d.). Retrieved August 6, 2011, from http://opende.sourceforge.net/wiki/index.php/Manual

Oliveira, D.M., et al. (2007). Virtual Reality System for Industrial Training. Industrial Electronics, International Symposium on , vol., no., pp.1715-1720, 4-7.

Rodrigues, F. L. D. (2010).Sistema de realidade virtual para simulador visual de passadiço (Virtual reality system for visual bridge simulator). (Master's Thesis, Escola Politécnica da Universidade de São Paulo).

Sadagic, A. (2010). Deconstructing game-based systems. Tutorial at *Interservice/Industry Training, Simulation and Education Conference (I/ITSEC 2010)*.

Salvatore, R. B. (2005). Using Open Source Software in Visual Simulation Development. (Master's Thesis, Naval Postgraduate School, Monterey, CA).

Ship Simulator. (n.d.). Retrieved May 13, 2011, from http://www.shipsim.com/products/shipsimulatorextremes

Souza, I. (2007). Simulador de Realidade Virtual para o Treinamento de Biópsia por Agulha de Nódulos da Glândula de Tireóide (Virtual Reality Simulator for Training Needle Biopsy of Thyroid Gland Nodules). (PhD Dissertation, Escola Politécnica da Universidade de São Paulo).

SurfTacs. (n.d.). Retrieved May 13, 2011, from http://www.delta3d.org/article.php?story=20050811204934496&topic=projects

Toledo, R. (2006). Design of an Object Oriented and Modular Architecture for a Naval Tactical Simulator Using Delta3D's Game Manager. (Master's Thesis, Naval Postgraduate School, Monterey, CA).

USNA (1991). *YP standard operating procedures (676 class)*. Annapolis, Maryland: USNA.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Secretaria de Ciência e Tecnologia da Marinha
   Brasília, Distrito Federal - Brazil

3. Centro de Análises de Sistemas Navais
   Centro, Rio de Janeiro - Brazil

4. Brazilian Naval Academy
   Castelo, Rio de Janeiro – Brazil

5. Brazilian Navy Attaché Office
   Brazilian Embassy
   Rockville, Maryland

6. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California